Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization

Daniel Golovin DGOLOVIN@CALTECH.EDU
California Institute of Technology
Pasadena, CA 91125, USA

Andreas Krause KRAUSEA@CALTECH.EDU
California Institute of Technology
Pasadena, CA 91125, USA

Abstract

Solving stochastic optimization problems under partial observability, where one needs to adaptively make decisions with uncertain outcomes, is a fundamental but notoriously difficult challenge. In this paper, we introduce the concept of *adaptive submodularity*, generalizing submodular set functions to adaptive policies. We prove that if a problem satisfies this property, a simple adaptive greedy algorithm is guaranteed to be competitive with the optimal policy. In addition to providing performance guarantees for both stochastic maximization and coverage, adaptive submodularity can be exploited to drastically speed up the greedy algorithm by using lazy evaluations. We illustrate the usefulness of the concept by giving several examples of adaptive submodular objectives arising in diverse applications including sensor placement, viral marketing and active learning. Proving adaptive submodularity for these problems allows us to recover existing results in these applications as special cases, improve approximation guarantees and handle natural generalizations.

Keywords: Adaptive Optimization, Stochastic Optimization, Submodularity, Partial Observability, Active Learning, Optimal Decision Trees

1. Introduction

In many practical optimization problems one needs to adaptively make a sequence of decisions, taking into account observations about the outcomes of past decisions. Often these outcomes are uncertain, and one may only know a probability distribution over them. Finding optimal policies for decision making in such partially observable stochastic optimization problems is notoriously intractable (see, e.g., Littman et al. (1998)). In this paper, we introduce the concept of *adaptive submodularity*, and prove that if a partially observable stochastic optimization problem satisfies this property, a simple adaptive greedy algorithm is guaranteed to obtain near-optimal solutions. Adaptive submodularity generalizes the notion of submodularity[1], which has been successfully used to develop approximation algorithms for a variety of non-adaptive optimization problems. Submodularity, informally, is an intuitive notion of diminishing returns, which states that adding an element to a small set helps more than adding that same element to a larger (super-)set. A celebrated result of Nemhauser et al. (1978) guarantees that for such submodular functions, a simple greedy algorithm, which adds the element that maximally increases the objective value, selects a near-optimal set of k elements. Similarly, it is guaranteed find a set of near-minimal cost that achieves a desired quota of utility (Wolsey, 1982), using near-minimum average time to do so (Streeter and Golovin, 2008). Besides guaranteeing theoretical performance bounds, submodularity allows us to speed up algorithms by using lazy evaluations (Minoux, 1978), often leading to performance improvements of several orders of magnitude (Leskovec et al., 2007). The challenge in generalizing submodularity to adaptive planning is that feasible solutions are now policies (decision trees) instead of subsets. We propose a natural generalization of the diminishing returns property for adaptive problems, which

0. An extended abstract of this work appeared in COLT 2010 (Golovin and Krause, 2010)
1. For an extensive treatment of submodularity, see the books of Fujishige (1991) and Schrijver (2003).

reduces to the classical characterization of submodular set functions for deterministic distributions. We show how these results of Nemhauser et al., Wolsey, Streeter and Golovin and Minoux generalize to the adaptive setting. Hence, we demonstrate how adaptive submodular optimization problems enjoy similar theoretical and practical benefits of classical, nonadaptive submodular problems. We further demonstrate the usefulness and generality of the concept by showing how it captures known results in stochastic optimization and active learning as special cases, admits tighter performance bounds, and leads to natural generalizations.

As a first example, consider the problem of deploying a collection of sensors to monitor some spatial phenomenon. Each sensor can cover a region depending on its sensing range. Suppose we would like to find the best subset of k locations to place the sensors. In this application, intuitively, adding a sensor helps more if we have placed few sensors so far and helps less if we have already placed many sensors. We can formalize this diminishing returns property using the notion of submodularity – the total area covered by the sensors is a submodular function defined over all sets of locations. Krause and Guestrin (2007) show that many more realistic utility functions in sensor placement (such as the improvement in prediction accuracy w.r.t. some probabilistic model) are submodular as well. Now consider the following stochastic variant: Instead of deploying a fixed set of sensors, we deploy one sensor at a time. With a certain probability, deployed sensors can fail, and our goal is to maximize the area covered by the functioning sensors. Thus, when deploying the next sensor, we need to take into account which of the sensors we deployed in the past failed. This problem has been studied by Asadpour et al. (2008) for the case where each sensor fails independently at random. In this paper, we show that the coverage objective is adaptive submodular, and use this concept to handle more general settings (where, e.g., rather than all-or-nothing failures there are different types of sensor failures of varying severity). We also consider the related problem of placing the minimum number of sensors to achieve the maximum possible sensor coverage (i.e., the coverage obtained by deploying sensors everywhere). This problem is equivalent to one studied by Goemans and Vondrák (2006), and generalizes a problem studied by Liu et al. (2008). As with the maximum coverage version, adaptive submodularity allows us to recover and generalize previous results.

As another example, consider a viral marketing problem, where we are given a social network, and we want to influence as many people as possible in the network to buy some product. We do that by giving the product for free to a subset of the people, and hope that they convince their friends to buy the product as well. Formally, we have a graph, and each edge e is labeled by a number $0 \leq p_e \leq 1$. We "influence" a subset of nodes in the graph, and for each influenced node, their neighbors get randomly influenced according to the probability annotated on the edge connecting the nodes. This process repeats until no further node gets influenced. Kempe et al. (2003) show that the set function which quantifies the expected number of nodes influenced is submodular. A natural stochastic variant of the problem is where we pick a node, get to see which nodes it influenced, then adaptively pick the next node based on these observations and so on. We show that a large class of such adaptive influence maximization problems satisfies adaptive submodularity.

Our third application is in pool-based active learning, where we are given an unlabeled data set, and we would like to adaptively pick a small set of examples whose labels imply all other labels. Thus, we want to pick examples to shrink the remaining version space (the set of consistent hypotheses) as quickly as possible. Here, we show that the reduction in version space probability mass is adaptive submodular, and use that observation to prove that the adaptive greedy algorithm is a near-optimal querying policy, recovering and generalizing results by Kosaraju et al. (1999) and Dasgupta (2004). Our results for active learning are also related to recent results of Guillory and Bilmes (2010) who study a generalization of submodular set cover to an interactive setting. In contrast to our approach however, Guillory and Bilmes analyze worst-case costs, and use rather different technical definitions and proof techniques.

We summarize our main contributions below, and provide a more technical summary in Table 1. At a high level, our main contributions are:

- We consider a particular class of partially observable adaptive stochastic optimization problems, which we prove to be hard to approximate in general.

Name	New Results
A.S. Maximization	Tight $(1 - 1/e)$-approx. for adaptive monotone submodular objectives, §5.1
A.S. Min Cost Coverage	Tight logarithmic approx. for adaptive monotone submodular objectives, §5.2
A.S. Min Sum Cover	Tight 4-approx. for adaptive monotone submodular objectives, §5.3
Data Dependent Bounds	Generalization of the data-dependent bounds for submodular functions, §5.1
Accelerated Adapt. Greedy	Generalization of lazy evaluations to the adaptive setting, §4
Stochastic Submodular Maximization	Generalization of the previous $(1 - 1/e)$-approx. to arbitrary per-item set distributions, and to item costs, §6
Stochastic Set Cover	Generalization of the previous $(\ln(n) + 1)$-approx. to arbitrary per-item set distributions, with item costs, §7
Adaptive Viral Marketing	Adaptive analog of previous $(1 - 1/e)$-approx. for non-adaptive viral marketing, under more general reward functions; tight logarithmic approx. for the adaptive min cost cover version, §8
Active Learning	Improved approx. factor of generalized binary search and its approximate versions with and without item costs, §9
Hardness in the absence of Adaptive Submodularity	$\Omega(\lvert E\rvert^{1-\epsilon})$-approximation hardness for A.S. Maximization, Min Cost Coverage, and Min-Sum Cover, if f is not adaptive submodular. §12

Table 1: Summary of our theoretical results. A.S. is shorthand for "Adaptive Stochastic".

- We introduce the concept of *adaptive submodularity*, and prove that if a problem instance satisfies this property, a simple adaptive greedy policy performs near-optimally, for both adaptive stochastic maximization and coverage, and also a natural min-sum objective.
- We show how adaptive submodularity can be exploited by allowing the use of an accelerated adaptive greedy algorithm using lazy evaluations, and how we can obtain tight, data-dependent bounds.
- We illustrate adaptive submodularity on several realistic problems, including Stochastic Maximum Coverage, Stochastic Submodular Coverage, Adaptive Viral Marketing, and Active Learning. For these applications, adaptive submodularity allows us to recover known results and prove natural generalizations.

Organization. This article is organized as follows. In §2 we set up notation and formally define the relevant adaptive optimization problems for general objective functions. For the reader's convenience, we have also provided a reference table of important symbols on page 46. In §3 we review the classical notion of submodularity and introduce the novel adaptive submodularity property. In §4 we introduce the adaptive greedy policy, as well as an accelerated variant. In §5 we discuss the theoretical guarantees that the adaptive greedy policy enjoys when applied to problems with adaptive submodular objectives. Sections 6 through 9 provide examples on how to apply the adaptive submodular framework to various applications, namely Stochastic Submodular Maximization (§6), Stochastic Submodular Coverage (§7), Adaptive Viral Marketing (§8), and Pool-based Active Learning (§9). In §10 we report empirical results on two sensor selection problems. In §11 we discuss the adaptivity gap of the problems we consider, and in §12 we prove hardness results indicating that problems which are not adaptive submodular can be extremely inapproximable under reasonable complexity assumptions. We review related work in §13 and provide concluding remarks in §14. The Appendix gives details of how to incorporate item costs and includes all of the proofs omitted from the main text.

2. Adaptive Stochastic Optimization

We start by introducing notation and defining the general class of adaptive optimization problems that we address in this paper.

$\pi(\emptyset) = a$
$\pi(\{(a,0)\}) = b$
$\pi(\{(a,1)\}) = c$
$\pi(\{(a,0),(b,0)\}) = d$
$\pi(\{(a,0),(b,1)\}) = e$
$\pi(\{(a,1),(c,1)\}) = f$

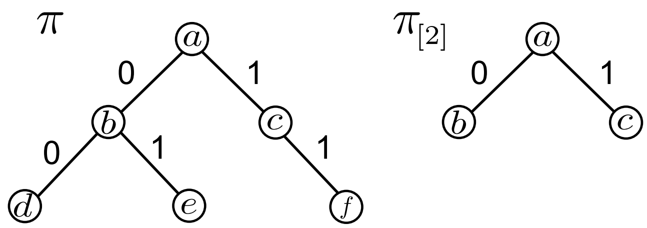


Figure 1: Illustration of a policy π, its corresponding decision tree representation, and the decision tree representation of $\pi_{[2]}$, the level 2 truncation of π (as defined in §5.1).

Items and Realizations. Let E be a finite set of items. Each item $e \in E$ is in a particular (initially unknown) state $\Phi(e) \in O$ from a set O of possible states. Hereby, $\Phi : E \to O$ is a (random) *realization* of the ground set, indicating which state each item is in. We take a Bayesian approach and assume that there is a (known) probability distribution $\mathbb{P}[\Phi]$ over realizations. We will consider problems where we sequentially pick an item $e \in E$, get to see its state $\Phi(e)$, pick the next item, get to see its state, and so on. After each pick, our observations so far can be represented as a *partial realization* Ψ, a function from some subset of E (i.e., the set of items that we already picked) to their states. For notational convenience, we sometimes represent Ψ as a relation, so that $\Psi \subseteq E \times O$ equals $\{(e,o) : \Psi(e) = o\}$. We use the notation $\mathrm{dom}(\Psi) = \{e : \exists o.(e,o) \in \Psi\}$ to refer to the domain of Ψ (i.e., the set of items observed in Ψ). A partial realization Ψ is *consistent* with a realization Φ if they are equal everywhere in the domain of Ψ. In this case we write $\Phi \sim \Psi$. If Ψ and Ψ' are both consistent with some Φ, and $\mathrm{dom}(\Psi) \subseteq \mathrm{dom}(\Psi')$, we say Ψ is a *subrealization* of Ψ'. Equivalently, Ψ is a subrealization of Ψ' if and only if $\Psi \subseteq \Psi'$.

Policies. We encode our adaptive strategy for picking items as a *policy* π, which is a function from a set of partial realizations to E, specifying which item to pick next under a particular set of observations. If $\Psi \notin \mathrm{dom}(\pi)$, the policy terminates (stops picking items) upon observation of Ψ. Technically, we require that the domain of π is closed under subrealizations. That is, if $\Psi' \in \mathrm{dom}(\pi)$ and Ψ is a subrealization of Ψ' then $\Psi \in \mathrm{dom}(\pi)$. We also allow randomized policies that are functions from a set of partial realizations to distributions on E. We use the notation $E(\pi, \Phi)$ to refer to the set of items selected by π under realization Φ. Each deterministic policy π can be associated with a decision tree T^π in a natural way (see Fig. 1 for an illustration). Here, we adopt a policy-centric view that admits concise notation, though we find the decision tree view to be valuable conceptually.

Adaptive Stochastic Maximization, Coverage, and Min-Sum Coverage. We wish to maximize, subject to some constraints, a utility function $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$ that depends on which items we pick and which state each item is in. Based on this notation, the expected utility of a policy π is $f_{\text{avg}}(\pi) := \mathbb{E}[f(E(\pi,\Phi),\Phi)]$ where the expectation is taken with respect to $\mathbb{P}[\Phi]$. The goal of the *Adaptive Stochastic Maximization* problem is to find a policy π^* such that

$$\pi^* \in \arg\max_{\pi} f_{\text{avg}}(\pi) \text{ subject to } |E(\pi,\Phi)| \leq k \text{ for all } \Phi, \tag{1}$$

where k is a budget on how many items can be picked.

Alternatively, we can specify a quota Q of utility that we would like to obtain, and try to find the cheapest policy achieving that quota. Formally, we define the average cost $c_{\text{avg}}(\pi)$ of a policy as the expected number of items it picks, so that $c_{\text{avg}}(\pi) := \mathbb{E}[|E(\pi,\Phi)|]$. Our goal is then to find

$$\pi^* \in \arg\min_{\pi} c_{\text{avg}}(\pi) \text{ such that } f(E(\pi,\Phi),\Phi) \geq Q \text{ for all } \Phi, \tag{2}$$

i.e., the policy π^* that minimizes the expected number of items picked such that under all possible realizations, at least utility Q is achieved. We call Problem 2 the *Adaptive Stochastic Minimum Cost Cover* problem. We

will also consider the problem where we want to minimize the worst-case cost $c_{\text{wc}}(\pi) := \max_\Phi |E(\pi, \Phi)|$. This worst-case cost $c_{\text{wc}}(\pi)$ is the cost incurred under adversarially chosen realizations, or equivalently the depth of the deepest leaf in T^π, the decision tree associated with π.

Yet another important variant is to minimize the average time required by a policy to obtain its utility. Formally, let $u(\pi, t)$ be the expected utility obtained by π after t steps, let $Q = \mathbb{E}[f(E, \Phi)]$ be the maximum possible expected utility, and define the *min-sum cost* $c_\Sigma(\pi)$ of a policy as $c_\Sigma(\pi) := \sum_{t=0}^{\infty} (Q - u(\pi, t))$. We then define the *Adaptive Stochastic Min-Sum Cover* problem as the search for

$$\pi^* \in \arg\min_\pi c_\Sigma(\pi)\,. \tag{3}$$

Unfortunately, as we will show in §12, even for linear functions f, i.e., those where $f(A, \Phi) = \sum_{e \in A} w_{e,\Phi}$ is simply the sum of weights (depending on the realization Φ), Problems (1), (2), and (3) are hard to approximate under reasonable complexity theoretic assumptions. Despite the hardness of the general problems, in the following sections we will identify conditions that are sufficient to allow us to approximately solve them.

Incorporating Item Costs. Instead of quantifying the cost of a set $E(\pi, \Phi)$ by the number of elements $|E(\pi, \Phi)|$, we can also consider the case where each item $e \in E$ has a cost $c(e)$, and the cost of a set $S \subseteq E$ is $c(S) = \sum_{e \in S} c(e)$. We can then consider variants of Problems (1), (2), and (3) with $|E(\pi, \Phi)|$ replaced by $c(E(\pi, \Phi))$. For clarity of presentation, we will focus on the unit cost case, i.e., $c(e) = 1$ for all e, and explain how our results generalize to the non-uniform case in the Appendix.

3. Adaptive Submodularity

We first review the classical notion of submodular set functions, and then introduce the novel notion of adaptive submodularity.

3.1 Background on Submodularity

Let us first consider the simple special case where $\mathbb{P}[\Phi]$ is deterministic or, equivalently, $|O| = 1$. In this case, the realization Φ is known to the decision maker in advance, and thus there is no benefit in adaptive selection. Thus, Problem (1) is equivalent to finding a set $A^* \subseteq E$ such that

$$A^* \in \arg\max_{A \subseteq E} f(A, \Phi) \text{ such that } |A| \leq k. \tag{4}$$

For most interesting classes of utility functions f, this is an NP-hard optimization problem. However, in many practical problems, such as those mentioned in §1, $f(A) = f(A, \Phi)$ satisfies *submodularity*. A set function $f : 2^E \to \mathbb{R}$ is called submodular if, whenever $A \subseteq B \subseteq E$ and $e \in E \setminus B$ it holds that

$$f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B), \tag{5}$$

i.e., adding e to the smaller set A increases f by at least as much as adding e to the superset B. Furthermore, f is called *monotone*, if, whenever $A \subseteq B$ it holds that $f(A) \leq f(B)$. A celebrated result by Nemhauser et al. (1978) states that for monotone submodular functions with $f(\emptyset) = 0$, a simple greedy algorithm that starts with the empty set, $A_0 = \emptyset$ and chooses

$$A_{i+1} = A_i \cup \{\arg\max_{e \in E \setminus A_i} f(A_i \cup \{e\})\} \tag{6}$$

guarantees that $f(A_k) \geq (1 - 1/e) \max_{|A| \leq k} f(A)$. Thus, the greedy set A_k obtains at least a $(1 - 1/e)$ fraction of the optimal value achievable using k elements. Furthermore, Feige (1998) shows that this result is tight if P $\neq$ NP; under this assumption no polynomial time algorithm can achieve a $(1 - 1/e + \epsilon)$-approximation for any constant $\epsilon > 0$, even for the special case of Maximum k-Cover where $f(A)$ is the cardinality of the union

of sets indexed by A. Similarly, Wolsey (1982) shows that the same greedy algorithm also near-optimally solves the deterministic case of Problem (2), called the *Minimum Submodular Cover* problem:

$$A^* \in \operatorname*{arg\,min}_{A \subseteq E} |A| \text{ such that } f(A) \geq Q. \tag{7}$$

Pick the first set A_ℓ constructed by the greedy algorithm such that $f(A_\ell) \geq Q$. Then, for integer-valued submodular functions, ℓ is at most $|A^*|(1 + \log(\max_e f(e)))$, i.e., the greedy set is at most a logarithmic factor larger than the smallest set achieving quota Q. For the special case of Set Cover, where $f(A)$ is the cardinality of a union of sets indexed by A, this result matches a lower bound by Feige (1998): Unless NP $\subseteq$ DTIME($n^{\mathcal{O}(\log \log n)}$), Set Cover is hard to approximate by a factor better than $(1 - \varepsilon) \ln Q$, where Q is the number of elements to be covered.

Now let us relax the assumption that $\mathbb{P}[\Phi]$ is deterministic. In this case, we may still want to find a non-adaptive solution (i.e., a constant policy π_A that always picks set A independently of Φ) maximizing $f_{\text{avg}}(\pi_A)$. If f is *pointwise* submodular, i.e., $f(A, \Phi)$ is submodular in A for any fixed Φ, the function $f(A) = f_{\text{avg}}(\pi_A)$ is submodular, since nonnegative linear combinations of submodular functions remain submodular. Thus, the greedy algorithm allows us to find a near-optimal *non-adaptive* policy.

However, in practice, we may be more interested in obtaining a non-constant policy π, that *adaptively* chooses items based on previous observations. Thus, the question is whether there is a natural extension of submodularity to policies. In the following, we will develop such a notion – *adaptive submodularity*.

3.2 Adaptive Monotonicity and Submodularity

The key challenge is to find appropriate generalizations of monotonicity and of the diminishing returns condition (5). We begin by considering the simple special case where $\mathbb{P}[\Phi]$ is deterministic, so that the policies are non-adaptive. In this case a policy π simply specifies a sequence of items $(e_1, e_2, \ldots, e_r)$ which it selects in order. Monotonicity in this context can be characterized as the property that "the marginal benefit of selecting an item is always nonnegative," meaning that for all such sequences $(e_1, e_2, \ldots, e_r)$, items e and $1 \leq i \leq r$ it holds that $f(\{e_j : j \leq i\} \cup \{e\}) - f(\{e_j : j \leq i\}) \geq 0$. Similarly, submodularity can be viewed as the property that "selecting an item later never increases its marginal benefit," meaning that for all sequences $(e_1, e_2, \ldots, e_r)$, items e, and all $i \leq r$, $f(\{e_j : j \leq i\} \cup \{e\}) - f(\{e_j : j \leq i\}) \geq f(\{e_j : j \leq r\} \cup \{e\}) - f(\{e_j : j \leq r\})$.

We take these views of monotonicity and submodularity when defining their adaptive analogues, by using an appropriate generalization of the marginal benefit. When moving to the general adaptive setting, the challenge is that for the items' states are now random and only revealed upon selection. A natural approach is thus to condition on observations (i.e., partial realizations of selected items), and take the expectation with respect to the items that we consider selecting. Hence, we define our adaptive monotonicity and submodularity properties in terms of the *conditional expected marginal benefit* of an item.

Definition 1 (Conditional Expected Marginal Benefit) *Given a partial realization Ψ and an item e, the* conditional expected marginal benefit *of e conditioned on having observed Ψ, denoted $\Delta(e \mid \Psi)$, is*

$$\Delta(e \mid \Psi) := \mathbb{E}\left[f(\operatorname{dom}(\Psi) \cup \{e\}, \Phi) - f(\operatorname{dom}(\Psi), \Phi) \mid \Phi \sim \Psi\right] \tag{8}$$

where the expectation is taken with respect to $\mathbb{P}[\Phi]$. Similarly, the conditional expected marginal benefit of a policy π *is*

$$\Delta(\pi \mid \Psi) := \mathbb{E}\left[f(\operatorname{dom}(\Psi) \cup E(\pi, \Phi), \Phi) - f(\operatorname{dom}(\Psi), \Phi) \mid \Phi \sim \Psi\right] \tag{9}$$

We are now ready to introduce our generalizations of monotonicity and submodularity to the adaptive setting:

Definition 2 (Adaptive Monotonicity) *A function $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$ is* adaptive monotone *with respect to distribution $\mathbb{P}[\Phi]$ if the conditional expected marginal benefit of any item is nonnegative, i.e., for all Ψ with $\mathbb{P}[\Psi] > 0$ and all $e \in E$ we have*

$$\Delta(e \mid \Psi) \geq 0. \tag{10}$$

Definition 3 (Adaptive Submodularity) *A function* $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$ *is* adaptive submodular *with respect to distribution* $\mathbb{P}[\Phi]$ *if the conditional expected marginal benefit of any fixed item does not increase as more items are selected and their states are observed. Formally, f is adaptive submodular w.r.t.* $\mathbb{P}[\Phi]$ *if for all* Ψ *and* Ψ' *such that* Ψ *is a subrealization of* Ψ' *(i.e.,* $\Psi \subseteq \Psi'$*), and for all* $e \in E$*, we have*

$$\Delta(e \,|\, \Psi') \leq \Delta(e \,|\, \Psi)\,. \tag{11}$$

From the decision tree perspective, the condition $\Delta(e \,|\, \Psi') \leq \Delta(e \,|\, \Psi)$ amounts to saying that for any decision tree T, if we are at a node v in T which selects an item e, and compare the expected marginal benefit of e selected at v with the expected marginal benefit e would have obtained if it were selected at an ancestor of v in T, then the latter must be no smaller than the former. Note that when comparing the two expected marginal benefits, both the set of items previously selected, and the distribution over realizations, are different. It is also worth stressing that adaptive submodularity is defined relative to the distribution $\mathbb{P}[\Phi]$ over realizations; it is possible that f is adaptive submodular with respect to one distribution, but not with respect to another.

We will give concrete examples of adaptive monotone and adaptive submodular functions that arise in the applications introduced in §1 in §6, §7, §8, and §9. In the Appendix, we will explain how the notion of adaptive submodularity can be extended to handle non-uniform costs.

Properties of Adaptive Submodular Functions. It can be seen that adaptive monotonicity and adaptive submodularity enjoy similar closure properties as monotone submodular functions. In particular, if $w_1, \ldots, w_m \geq 0$ and $f_1, \ldots, f_m$ are adaptive monotone submodular w.r.t. distribution $\mathbb{P}[\Phi]$, then $f(A, \Phi) = \sum_{i=1}^m w_i f_i(A, \Phi)$ is adaptive monotone submodular w.r.t. $\mathbb{P}[\Phi]$. Similarly, for a fixed constant $c \geq 0$ and adaptive monotone submodular function f, the function $g(E, \Phi) = \min(f(E, \Phi), c)$ is adaptive monotone submodular. Thus, adaptive monotone submodularity is preserved by nonnegative linear combinations and by truncation. Adaptive monotone submodularity is also preserved by restriction, so that if $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$ is adaptive monotone submodular w.r.t. $\mathbb{P}[\Phi]$, then for any $e \in E$, the function $g : 2^{E \setminus \{e\}} \times O^E \to \mathbb{R}_{\geq 0}$ defined by $g(A, \Phi) := f(A, \Phi)$ for all $A \subseteq E \setminus \{e\}$ and all Φ is also adaptive submodular w.r.t. $\mathbb{P}[\Phi]$. Finally, if $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$ is adaptive monotone submodular w.r.t. $\mathbb{P}[\Phi]$ then for each partial realization Ψ the conditional function $g(A, \Phi) := f(A \cup \mathrm{dom}(\Psi), \Phi)$ is adaptive monotone submodular w.r.t. $\mathbb{P}[\Phi \mid \Psi]$.

4. The Adaptive Greedy Policy

The classical non-adaptive greedy algorithm (6) has a natural generalization to the adaptive setting. The *greedy policy* π^{greedy} tries, at each iteration, to myopically increase the expected objective value, given its current observations. That is, suppose $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$ is the objective, and Ψ is the partial realization indicating the states of items selected so far. Then the greedy policy will select the item e maximizing the expected increase in value, conditioned on the observed states of items it has already selected (i.e., conditioned on $\Phi \sim \Psi$). That is, it will select e to maximize the conditional expected marginal benefit $\Delta(e \,|\, \Psi)$ as defined in Eq. (8). Pseudocode of the adaptive greedy algorithm is given in Algorithm 1. The only difference to the classic, non-adaptive greedy algorithm studied by Nemhauser et al. (1978), is Line 1, where an observation $\Phi(e^*)$ of the selected item e^* is obtained. Note that the algorithms in this section are presented for Adaptive Stochastic Maximization. For the coverage objectives, we simply keep selecting items as prescribed by π^{greedy} until achieving the quota on objective value (for the min-cost objective) or until we have selected every item (for the min-sum objective).

Incorporating Item Costs. The adaptive greedy algorithm can be naturally modified to handle non-uniform item costs by replacing its selection rule by

$$e^* \in \arg\max_e \frac{\Delta(e \,|\, \Psi)}{c(e)}.$$

In the following, we will focus on the uniform cost case ($c \equiv 1$), and defer the analysis with costs to the Appendix.

Approximate Greedy Selection. In some applications, finding an item maximizing $\Delta(e \mid \Psi)$ may be computationally intractable, and the best we can do is find an α-approximation to the best greedy selection. This means we find an e' such that

$$\Delta(e' \mid \Psi) \geq \frac{1}{\alpha} \max_e \Delta(e \mid \Psi) .$$

We call a policy which always selects such an item an *α-approximate greedy policy*.

Input: Budget k; ground set E; distribution $\mathbb{P}[\Phi]$; function f.
Output: Set $A \subseteq E$ of size k
begin
  $A \leftarrow \emptyset$; $\Psi \leftarrow \emptyset$;
  for $i = 1$ **to** k **do**
    foreach $e \in E \setminus A$ **do** compute $\Delta(e \mid \Psi) = \mathbb{E}[f(A \cup \{e\}, \Phi) - f(A, \Phi) \mid \Phi \sim \Psi]$;
    Select $e^* \in \arg\max_e \Delta(e \mid \Psi)$;
    Set $A \leftarrow A \cup \{e^*\}$;
    Observe $\Phi(e^*)$; Set $\Psi \leftarrow \Psi \cup \{(e^*, \Phi(e^*))\}$;
end

Algorithm 1: The adaptive greedy algorithm, which implements the greedy policy.

Lazy Evaluations and the Accelerated Adaptive Greedy Algorithm. The definition of adaptive submodularity allows us to implement an "accelerated" version of the adaptive greedy algorithm using lazy evaluations of marginal benefits as originally suggested for the non-adaptive case by Minoux (1978). The idea is as follows. Suppose we run π^{greedy} under some fixed realization Φ, and select items $e_1, e_2, \ldots, e_k$. Let $\Psi_i := \{(e_j, \Phi(e_j) : j \leq i)\}$ be the partial realizations observed during the run of π^{greedy}. The adaptive greedy algorithm computes $\Delta(e \mid \Psi_i)$ for all $e \in E$ and $0 \leq i < k$, unless $e \in \mathrm{dom}(\Psi_i)$. Naively, the algorithm thus needs to compute $\Theta(|E|k)$ marginal benefits (which can be expensive to compute). The key insight is that $i \mapsto \Delta(e \mid \Psi_i)$ is nonincreasing for all $e \in E$, because of the adaptive submodularity of the objective. Hence, if when deciding which item to select as e_i we know $\Delta(e' \mid \Psi_j) \leq \Delta(e \mid \Psi_i)$ for some items e' and e and $j < i$, then we may conclude $\Delta(e' \mid \Psi_i) \leq \Delta(e \mid \Psi_i)$ and hence eliminate the need to compute $\Delta(e' \mid \Psi_i)$. The accelerated version of the adaptive greedy algorithm exploits this observation in a principled manner, by computing $\Delta(e \mid \Psi)$ for items e in decreasing order of the upper bounds known on them, until it finds an item whose value is at least as great as the upper bounds of all other items. Pseudocode of this version of the adaptive greedy algorithm is given in Algorithm 2.

In the non-adaptive setting, the use of lazy evaluations has been shown to significantly reduce running times in practice (Leskovec et al., 2007). We evaluated the naive and accelerated implementations of the adaptive greedy algorithm on two sensor selection problems, and obtained speedup factors that range from roughly 4 to 40 for those problems. See §10 for details.

5. Guarantees for the Greedy Policy

In this section we show that if the objective function is adaptive submodular with respect to the distribution describing the environment in which we operate, then the greedy policy and any α-approximate greedy policy inherit precisely the performance guarantees of the greedy and α-approximate greedy algorithms for classic (non-adaptive) submodular maximization and submodular coverage problems, such as Maximum k-Cover and Minimum Set Cover, as well as min-sum submodular coverage problems, such as Min-Sum Set Cover. These guarantees suggest that adaptive submodularity is the appropriate generalization of submodularity to policies. In this section we focus on the unit cost case. In the Appendix we provide the proofs omitted in this section, and show how our results extend to non-uniform item costs if we greedily maximize the expected benefit/cost ratio.

Input: Budget k; ground set E; distribution $\mathbb{P}[\Phi]$; function f.
Output: Set $A \subseteq E$ of size k
begin
  $A \leftarrow \emptyset$; $\Psi \leftarrow \emptyset$; Priority Queue $Q \leftarrow$ EMPTY_QUEUE;
  foreach $e \in E$ **do** $Q.\operatorname{insert}(e, +\infty)$;
  for $i = 1$ **to** k **do**
    $\delta_{\max} \leftarrow -\infty$; $e_{\max} \leftarrow$ NULL;
    while $\delta_{\max} < Q.\operatorname{maxPriority}(\,)$ **do**
      $e \leftarrow Q.\operatorname{pop}(\,)$;
      $\delta \leftarrow \Delta(e \,|\, \Psi) = \mathbb{E}\left[f(A \cup \{e\}, \Phi) - f(A, \Phi) \mid \Phi \sim \Psi\right]$;
      $Q.\operatorname{insert}(e, \delta)$;
      if $\delta_{\max} < \delta$ **then**
        $\delta_{\max} \leftarrow \delta$; $e_{\max} \leftarrow e$;
    $A \leftarrow A \cup \{e_{\max}\}$; $Q.\operatorname{remove}(e_{\max})$;
    Observe $\Phi(e_{\max})$; Set $\Psi \leftarrow \Psi \cup \{(e_{\max}, \Phi(e_{\max}))\}$;
end

Algorithm 2: The accelerated version of the adaptive greedy algorithm. Here, $Q.\operatorname{insert}(e, \delta)$ inserts e with priority δ, $Q.\operatorname{pop}(\,)$ removes and returns the item with greatest priority, $Q.\operatorname{maxPriority}(\,)$ returns the maximum priority of the elements in Q, and $Q.\operatorname{remove}(e)$ deletes e from Q.

5.1 The Maximum Coverage Objective

In this section we consider the maximum coverage objective, where the goal is to select k items adaptively to maximize their expected value. Before stating our result, we require the following definition.

Definition 4 (Policy Truncation) *For a policy π, define the* level-k-truncation $\pi_{[k]}$ *of π to be the policy obtained by running π until it terminates or until it selects k items, and then terminating. Formally,* $\operatorname{dom}(\pi_{[k]}) = \{\Psi \in \operatorname{dom}(\pi) \; : \; |\Psi| < k\}$, *and* $\pi_{[k]}(\Psi) = \pi(\Psi)$ *for all* $\Psi \in \operatorname{dom}(\pi_{[k]})$.

We have the following result, which generalizes the classic result of Nemhauser et al. (1978) on maximizing monotone submodular functions under a cardinality constraint.

Theorem 5 *Fix any $\alpha \geq 1$. If f is adaptive monotone and adaptive submodular with respect to the distribution $\mathbb{P}[\Phi]$, and π is an α-approximate greedy policy, then for all policies π^* and positive integers ℓ and k,*

$$f_{avg}(\pi_{[\ell]}) > \left(1 - e^{-\ell/\alpha k}\right) f_{avg}(\pi^*_{[k]}).$$

In particular, with $\ell = k$ this implies any α-approximate greedy policy achieves a $\left(1 - e^{-1/\alpha}\right)$ approximation to the expected reward of the best policy, if both are terminated after running for an equal number of steps.

If the greedy rule can be implemented only with small *absolute* error rather than small *relative* error, i.e., $\Delta(e' \,|\, \Psi) \geq \max_e \Delta(e \,|\, \Psi) - \varepsilon$, an argument similar to that used to prove Theorem 5 shows that

$$f_{\text{avg}}(\pi_{[\ell]}) \geq \left(1 - e^{-\ell/k}\right) f_{\text{avg}}(\pi^*_{[k]}) - \ell\varepsilon.$$

This is important, since small absolute error can always be achieved (with high probability) whenever f can be evaluated efficiently, and sampling $P(\Phi \mid \Psi)$ is efficient. In this case, we can approximate

$$\Delta(e \,|\, \Psi) \approx \frac{1}{N} \sum_{i=1}^{N} \left[f(\operatorname{dom}(\Psi) \cup \{e\}, \Phi_i) - f(\operatorname{dom}(\Psi), \Phi_i)\right],$$

where Φ_i are sampled i.i.d. from $P(\Phi \mid \Psi)$.

Data Dependent Bounds. For the maximum coverage objective, adaptive submodular functions have another attractive feature: they allow us to obtain data dependent bounds on the optimum, in a manner similar to the bounds for the non-adaptive case (Minoux, 1978). Consider the non-adaptive problem of maximizing a monotone submodular function $f : 2^A \to \mathbb{R}_{\geq 0}$ subject to the constraint $|A| \leq k$. Let A^* be an optimal solution, and fix any $A \subseteq E$. Then

$$f(A^*) \leq f(A) + \max_{B:|B| \leq k} \sum_{e \in B} (f(A \cup \{e\}) - f(A)) \tag{12}$$

because setting $B = A^*$ we have $f(A^*) \leq f(A \cup B) \leq f(A) + \sum_{e \in B} (f(A \cup \{e\}) - f(A))$. Note that unlike the original objective, we can easily compute $\max_{B:|B| \leq k} \sum_{e \in B} (f(A \cup \{e\}) - f(A))$ by computing $\delta(e) := f(A \cup \{e\}) - f(A)$ for each e, and summing the k largest values. Hence we can quickly compute an upper bound on our distance from the optimal value, $f(A^*) - f(A)$. In practice, such data-dependent bounds can be much tighter than the problem-independent performance guarantees of Nemhauser et al. for the greedy algorithm. Further note that these bounds hold for *any* set A, not just sets selected by the greedy algorithm.

These data dependent bounds have the following analogue for adaptive monotone submodular functions.

Lemma 6 (The Adaptive Data Dependent Bound) *Suppose we have made observations Ψ after selecting $\mathrm{dom}(\Psi)$. Let π^* be any policy such that $|E(\pi^*, \Phi)| \leq k$ for all Φ. Then for adaptive monotone submodular f*

$$\Delta(\pi^* \,|\, \Psi) \;\leq\; \max_{A \subseteq E, |A| \leq k} \sum_{e \in A} \Delta(e \,|\, \Psi)\,. \tag{13}$$

Thus, after running *any* policy π, we can efficiently compute a bound on the additional benefit that the optimal solution π^* could obtain beyond the reward of π. We do that by computing the conditional expected marginal benefits for all elements e, and summing the k largest of them. Note that these bounds can be computed on the fly when running the greedy algorithm.

5.2 The Min Cost Cover Objective

Another natural objective is to minimize the number of items selected while ensuring that a sufficient level of value is obtained. This leads to the *Adaptive Stochastic Minimum Cost Coverage* problem described in §2, namely $\pi^* \in \arg\min_\pi c_{\text{avg}}(\pi)$ such that $f(E(\pi, \Phi), \Phi) \;\geq\; Q$ for all Φ. Recall that $c_{\text{avg}}(\pi)$ is the expected cost of π, which in the unit cost case equals the expected number of items selected by π, i.e., $c_{\text{avg}}(\pi) := \mathbb{E}\left[|E(\pi, \Phi)|\right]$. If the objective is adaptive monotone submodular, this is an adaptive version of Minimum Submodular Cover (described on line (7) in §3.1), for which the greedy algorithm is known to give a $(\ln(Q) + 1)$-approximation for integer-valued monotone submodular functions (Wolsey, 1982). It is also related to the Interactive Submodular Set Cover problem studied by Guillory and Bilmes (2010), which considers the worst-case setting (i.e., there is no distribution over states; instead states are realized in an adversarial manner). Similar results for active learning have been proved by Kosaraju et al. (1999) and Dasgupta (2004), as we discuss in more detail in §9.

We assume throughout this section that there exists a quality threshold Q such that $f(E, \Phi) = Q$ for all Φ, and for all $S \subseteq E$ and all Φ, $f(S, \Phi) \leq Q$. Note that, as discussed in Section 3, if we replace $f(S, \Phi)$ by a new function $g(S, \Phi) = \min(f(S, \Phi), Q')$ for some constant Q', g will be adaptive submodular if f is. Thus, if $f(E, \Phi)$ varies across realizations, we can instead use the greedy algorithm on the function truncated at some threshold $Q' \leq \min_\Phi f(E, \Phi)$ achievable by all realizations.

In contrast to Adaptive Stochastic Maximization, for the coverage problem additional subtleties arise. In particular, it is not enough that a policy π achieves value Q for the true realization; in order for π to terminate, it also requires a proof of this fact. Formally, we require that π *covers* f:

Definition 7 (Coverage) *Let $\Psi = \Psi(\pi, \Phi)$ be the partial realization encoding all states observed during the execution of π under true realization Φ. Given $f : 2^E \times O^E \to \mathbb{R}$, we say a policy π* covers Φ *with respect to f if $f(\mathrm{dom}(\Psi), \Phi') = f(E, \Phi')$ for all $\Phi' \sim \Psi$. We say that π* covers f *if it covers every realization with respect to f.*

Coverage is defined in such a way that upon terminating, π might not know which realization in $\{\Phi' : \Phi' \sim \Psi(\pi, \Phi)\}$ is the true one, but has guaranteed that it has achieved the maximum reward in every possible case. We obtain results for both the average and worst-case cost objectives.

Minimizing the Average Cost. Before presenting our approximation guarantee for the Adaptive Stochastic Minimum Cost Coverage, we introduce a special class of instances, called *self–certifying* instances. We make this distinction because the greedy policy has stronger performance guarantees for self–certifying instances, and such instances arise naturally in applications. For example, the Stochastic Submodular Cover and Stochastic Set Cover instances in §7, the Adaptive Viral Marketing instances in §8, and the Pool-Based Active Learning instances in §9 are all self–certifying.

Definition 8 (Self–Certifying Instances) *An instance of Adaptive Stochastic Minimum Cost Coverage is* self–certifying *if whenever a policy achieves the maximum possible value for the true realization it immediately has a proof of this fact. Formally, an instance $(f, \mathbb{P}[\Phi])$ is self–certifying if for all Φ, Φ', and Ψ such that $\Phi \sim \Psi$ and $\Phi' \sim \Psi$, we have $f(\mathrm{dom}(\Psi), \Phi) = f(E, \Phi)$ if and only if $f(\mathrm{dom}(\Psi), \Phi') = f(E, \Phi')$.*

For minimum cost coverage, we also need a stronger monotonicity condition:

Definition 9 (Strong Adaptive Monotonicity) *A function $f : 2^E \times O^E \to \mathbb{R}$ is* strongly adaptive monotone *with respect to $\mathbb{P}[\cdot]$ if, informally "selecting more items never hurts" with respect to the expected reward. Formally, for all Ψ, all $e \notin \mathrm{dom}(\Psi)$, and all $o \in O$ such that $\mathbb{P}[\Phi(e) = o \mid \Phi \sim \Psi] > 0$, we require*

$$\mathbb{E}[f(\mathrm{dom}(\Psi), \Phi) \mid \Phi \sim \Psi] \le \mathbb{E}[f(\mathrm{dom}(\Psi) \cup \{e\}, \Phi) \mid \Phi \sim \Psi, \Phi(e) = o]. \tag{14}$$

Strong adaptive monotonicity implies adaptive monotonicity, as the latter means that "selecting more items never hurts in expectation," i.e., $\mathbb{E}[f(\mathrm{dom}(\Psi), \Phi) \mid \Phi \sim \Psi] \le \mathbb{E}[f(\mathrm{dom}(\Psi) \cup \{e\}, \Phi) \mid \Phi \sim \Psi]$. We now state our main result for the average case cost $c_{\mathrm{avg}}(\pi)$:

Theorem 10 *Suppose $f : 2^E \times O^E \to \mathbb{R}_{\ge 0}$ is adaptive submodular and strongly adaptive monotone with respect to $\mathbb{P}[\Phi]$ and there exists Q such that $f(E, \Phi) = Q$ for all Φ. Let η be any value such that $f(S, \Phi) > Q - \eta$ implies $f(S, \Phi) = Q$ for all S and Φ. Let $\delta = \min_\Phi \mathbb{P}[\Phi]$ be the minimum probability of any realization. Let π^*_{avg} be an optimal policy minimizing the expected number of items selected to guarantee every realization is covered. Let π be an α-approximate greedy policy. Then in general*

$$c_{avg}(\pi) \le \alpha\, c_{avg}(\pi^*_{avg}) \left(\ln\left(\frac{Q}{\delta\eta}\right) + 1 \right)$$

and for self–certifying instances

$$c_{avg}(\pi) \le \alpha\, c_{avg}(\pi^*_{avg}) \left(\ln\left(\frac{Q}{\eta}\right) + 1 \right).$$

Note that if $\mathrm{range}(f) \subset \mathbb{Z}$, then $\eta = 1$ is a valid choice, so in this case $c_{avg}(\pi) \le \alpha\, c_{avg}(\pi^*_{avg}) (\ln(Q/\delta) + 1)$ and $c_{avg}(\pi) \le \alpha\, c_{avg}(\pi^*_{avg}) (\ln(Q) + 1)$ for general and self–certifying instances, respectively.

Minimizing the Worst-Case Cost. For the worst-case cost $c_{\mathrm{wc}}(\pi) := \max_\Phi |E(\pi, \Phi)|$, strong adaptive monotonicity is not required; adaptive monotonicity suffices. We obtain the following result.

Theorem 11 *Suppose $f : 2^E \times O^E \to \mathbb{R}_{\ge 0}$ is adaptive monotone and adaptive submodular with respect to $\mathbb{P}[\Phi]$, and let η be any value such that $f(S, \Phi) > f(E, \Phi) - \eta$ implies $f(S, \Phi) = f(E, \Phi)$ for all S and Φ. Let $\delta = \min_\Phi \mathbb{P}[\Phi]$ be the minimum probability of any realization. Let π^*_{wc} be the optimal policy minimizing the worst-case number of queries to guarantee every realization is covered. Let π be an α-approximate greedy policy. Finally, let $Q := \mathbb{E}[f(E, \Phi)]$ be the maximum possible expected reward. Then*

$$c_{wc}(\pi) \le \alpha\, c_{wc}(\pi^*_{wc}) \left(\ln\left(\frac{Q}{\delta\eta}\right) + 1 \right).$$

The proofs of Theorems 10 and 11 are given in Appendix A.4.

Thus, even though adaptive submodularity is defined w.r.t. a particular distribution, perhaps surprisingly, the adaptive greedy algorithm is competitive even in the case of adversarially chosen realizations, against a policy optimized to minimize the worst-case cost. Theorem 11 therefore suggests that if we do not have a strong prior, we can obtain the strongest guarantees if we choose a distribution that is "as uniform as possible" (i.e., maximizes δ) while still guaranteeing adaptive submodularity.

Discussion. Note that the approximation factor for self–certifying instances in Theorem 10 reduces to the $(\ln(Q) + 1)$-approximation guarantee for the greedy algorithm for Set Cover instances with Q elements, in the case of a deterministic distribution $\mathbb{P}[\Phi]$. Moreover, with a deterministic distribution $\mathbb{P}[\Phi]$ there is no distinction between average-case and worst-case cost. Hence, an immediate corollary of the result of Feige (1998) mentioned in §3 is that for every constant $\epsilon > 0$ there is no polynomial time $(1 - \epsilon)\ln(Q/\eta)$ approximation algorithm for self–certifying instances of Adaptive Stochastic Min Cost Cover, under either the $c_{\text{avg}}(\cdot)$ or the $c_{\text{wc}}(\cdot)$ objective, unless NP $\subseteq$ DTIME($n^{\mathcal{O}(\log\log n)}$). It remains open to determine whether or not Adaptive Stochastic Min Cost Cover with the worst-case cost objective admits a $\ln(Q/\eta) + 1$ approximation for self–certifying instances via a polynomial time algorithm, and in particular whether the greedy policy has such an approximation guarantee. However, in Lemma 36 we show that Feige's result also implies there is no $(1 - \epsilon)\ln(Q/\delta\eta)$ polynomial time approximation algorithm for general (non self-certifying) instances of Adaptive Stochastic Min Cost Cover under either objective, unless NP $\subseteq$ DTIME($n^{\mathcal{O}(\log\log n)}$). In that sense, each of the three results comprising Theorem 10 and Theorem 11 are best-possible under reasonable complexity-theoretic assumptions. As we show in Section 9, our result for the average-case cost of greedy policies for self–certifying instances also matches (up to constant factors) results on hardness of approximating the optimal policy in the special case of active learning, also known as the *Optimal Decision Tree* problem.

5.3 The Min-Sum Cover Objective

Yet another natural objective is the *min-sum* objective, in which an unrealized reward of x incurs a cost of x in each time step, and the goal is to minimize the total cost incurred.

Background on the Non-adaptive Min-Sum Cover Problem. In the non-adaptive setting, perhaps the simplest form of a coverage problem with this objective is the *Min-Sum Set Cover* problem (Feige et al., 2004) in which the input is a set system $(U, \mathcal{S})$, the output is a permutation of the sets $\langle S_1, S_2, \ldots, S_m\rangle$, and the goal is to minimize the sum of element *coverage times*, where the coverage time of u is the index of the first set that contains it (e.g., it is j if $u \in S_j$ and $u \notin S_i$ for all $i < j$). In this problem and its generalizations the min-sum objective is useful in modeling processing costs in certain applications, for example in ordering diagnostic tests to identify a disease cheaply (Kaplan et al., 2005), in ordering multiple filters to be applied to database records while processing a query (Munagala et al., 2005), or in ordering multiple heuristics to run on boolean satisfiability instances as a means to solve them faster in practice (Streeter and Golovin, 2008). A particularly expressive generalization of min-sum set cover has been studied under the names *Min-Sum Submodular Cover* (Streeter and Golovin, 2008) and *L_1-Submodular Set Cover* (Golovin et al., 2008). The former paper extends the greedy algorithm to a natural online variant of the problem, while the latter studies a parameterized family of *L_p-Submodular Set Cover* problems in which the objective is analogous to minimizing the L_p norm of the coverage times for Min-Sum Set Cover instances. In the Min-Sum Submodular Cover problem, there is a monotone submodular function $f : 2^E \to \mathbb{R}_{\geq 0}$ defining the reward obtained from a collection of elements[2]. There is an integral cost $c(e)$ for each element, and the output is a sequence of all of the elements $\sigma = \langle e_1, e_2, \ldots, e_n\rangle$. For each $t \in \mathbb{R}_{\geq 0}$, we define the set of elements in the sequence σ within a budget of t:

$$\sigma_{[t]} := \left\{ e_i : \sum_{j \leq i} c(e_j) \leq t \right\}$$

2. To encode Min-Sum Set Cover instance $(U, \mathcal{S})$, let $E := \mathcal{S}$ and $f(A) := |\cup_{e \in A}\, e|$, where each $e \in E$ is a subset of elements in U.

The cost we wish to minimize is then

$$c_\Sigma(\sigma) := \sum_{t=0}^{\infty} \left(f(E) - f(\sigma_{[t]}) \right) \tag{15}$$

Feige et al. (2004) proved that for Min-Sum Set cover, the greedy algorithm achieves a 4-approximation to the minimum cost, and also that this is optimal in the sense that no polynomial time algorithm can achieve a $(4-\epsilon)$-approximation, for any $\epsilon > 0$, unless P = NP. Interestingly, the greedy algorithm also achieves a 4-approximation for the more general Min-Sum Submodular Cover problem as well (Streeter and Golovin, 2008; Golovin et al., 2008).

The Adaptive Stochastic Min-Sum Cover Problem. In this article, we extend the result of Streeter and Golovin (2008); Golovin et al. (2008) to an adaptive version of Min-Sum Submodular Cover. For clarity's sake we will consider the unit-cost case here (i.e., $c(e) = 1$ for all e); we show how to extend adaptive submodularity to handle general costs in the Appendix. In the adaptive version of the problem, $\pi_{[t]}$ plays the role of $\sigma_{[t]}$, and f_{avg} plays the role of f. The goal is to find a policy π minimizing

$$c_\Sigma(\pi) := \sum_{t=0}^{\infty} \left(\mathbb{E}\left[f(E,\Phi)\right] - f_{\text{avg}}(\pi_{[t]}) \right) = \sum_{\Phi} \mathbb{P}\left[\Phi\right] \sum_{t=0}^{\infty} \left(f(E,\Phi) - f(E(\pi_{[t]},\Phi),\Phi) \right) \tag{16}$$

We call this problem the *Adaptive Stochastic Min-Sum Cover* problem. The key difference between this objective and the minimum cost cover objective is that here, the cost at each step is only the fractional extent that we have not covered the true realization, whereas in the minimum cost cover objective we are charged in full in each step until we have completely covered the true realization (according to Definition 7). We prove the following result for the Adaptive Stochastic Min-Sum Cover problem with arbitrary item costs in Appendix A.5.

Theorem 12 *Fix any $\alpha \geq 1$. If f is adaptive monotone and adaptive submodular with respect to the distribution $\mathbb{P}[\Phi]$, π is an α-approximate greedy policy with respect to the item costs, and π^* is any policy, then $c_\Sigma(\pi) \leq 4\alpha\, c_\Sigma(\pi^*)$.*

6. Application: Stochastic Submodular Maximization

As our first application, consider the sensor placement problem introduced in §1. Suppose we would like to monitor a spatial phenomenon such as temperature in a building. We discretize the environment into a set E of locations. We would like to pick a subset $A \subseteq E$ of k locations that is most "informative", where we use a set function $\hat{f}(A)$ to quantify the informativeness of placement A. Krause and Guestrin (2007) show that many natural objective functions (such as reduction in predictive uncertainty measured in terms of Shannon entropy with conditionally independent observations) are monotone submodular.

Now consider the problem, where sensors can fail or partially fail (e.g., be subject to some varying amount of noise) after deployment. We can model this extension by assigning a state $\Phi(e) \in O$ to each possible location, indicating the extent to which a sensor placed at location e is working. To quantify the value of a set of sensor deployments under a realization Φ indicating to what extent the various sensors are working, we first define (e, o) for each $e \in E$ and $o \in O$, which represents the placement of a sensor at location e which is in state o. We then suppose there is a function $\hat{f} : 2^{E\times O} \to \mathbb{R}_{\geq 0}$ which quantifies the informativeness of a set of sensor deployments in arbitrary states. The utility $f(A, \Phi)$ of placing sensors at the locations in A under realization Φ is then

$$f(A,\Phi) := \hat{f}(\{(e,\Phi(e)) : e \in A\}).$$

We aim to adaptively place k sensors to maximize our expected utility. We assume that sensor failures at each location are independent of each other, i.e., $\mathbb{P}[\Phi] = \prod_e \mathbb{P}[\Phi(e)]$, where $\mathbb{P}[\Phi(e) = o]$ is the probability

that a sensor placed at location e will be in state o. Asadpour et al. (2008) studied a special case of our problem, in which sensors either fail completely (in which case they contribute no value at all) or work perfectly, under the name *Stochastic Submodular Maximization*. They proved that the adaptive greedy algorithm obtains a $(1 - 1/e)$ approximation to the optimal adaptive policy, provided $\hat{f}$ is monotone submodular. We extend their result to multiple types of failures by showing that $f(A, \Phi)$ is adaptive submodular with respect to distribution $\mathbb{P}[\Phi]$ and then invoking Theorem 5. Fig. 2 illustrates an instance of the Stochastic Submodular Maximization problem where $f(A, \Phi)$ is the cardinality of union of sets index by A and parameterized by Φ.

Theorem 13 *Fix a prior such that $\mathbb{P}[\Phi] = \prod_{e \in E} \mathbb{P}[\Phi(e)]$ and an integer k, and let the objective function $\hat{f} : 2^{E \times O} \to \mathbb{R}_{\geq 0}$ be monotone submodular. Let π be any α-approximate greedy policy attempting to maximize f, and let π^* be any policy. Then for all positive integers ℓ,*

$$f_{avg}(\pi_{[\ell]}) \geq \left(1 - e^{-\ell/\alpha k}\right) f_{avg}(\pi^*_{[k]}).$$

In particular, if π is the greedy policy (i.e., $\alpha = 1$) and $\ell = k$, then $f_{avg}(\pi_{[k]}) \geq \left(1 - \frac{1}{e}\right) f_{avg}(\pi^*_{[k]})$.

Proof We prove Theorem 13 by first proving f is adaptive monotone and adaptive submodular in this model, and then applying Theorem 5. Adaptive monotonicity is readily proved after observing that $f(\cdot, \Phi)$ is monotone for each Φ. Moving on to adaptive submodularity, fix any Ψ, Ψ' such that $\Psi \subseteq \Psi'$ and any $e \notin \mathrm{dom}(\Psi')$. We aim to show $\Delta(e \,|\, \Psi') \leq \Delta(e \,|\, \Psi)$. Intuitively, this is clear, as $\Delta(e \,|\, \Psi')$ is the expected marginal benefit of adding e to a larger base set than is the case with $\Delta(e \,|\, \Psi)$, namely $\mathrm{dom}(\Psi')$ as compared to $\mathrm{dom}(\Psi)$, and the realizations are independent. To prove it rigorously, we define a coupled distribution μ over pairs of realizations $\Phi \sim \Psi$ and $\Phi' \sim \Psi'$ such that $\Phi(e') = \Phi'(e')$ for all $e' \notin \mathrm{dom}(\Psi')$. Formally, $\mu(\Phi, \Phi') = \prod_{e \in E \setminus \mathrm{dom}(\Psi)} \mathbb{P}[\Phi(e)]$ if $\Phi \sim \Psi$, $\Phi' \sim \Psi'$, and $\Phi(e') = \Phi'(e')$ for all $e' \notin \mathrm{dom}(\Psi')$; otherwise $\mu(\Phi, \Phi') = 0$. (Note that $\mu(\Phi, \Phi') > 0$ implies $\Phi(e') = \Phi'(e')$ for all $e' \in \mathrm{dom}(\Psi)$ as well, since $\Phi \sim \Psi$, $\Phi' \sim \Psi'$, and $\Psi \subseteq \Psi'$.) Also note that $\mathbb{P}[\Phi \mid \Psi] = \sum_{\Phi'} \mu(\Phi, \Phi')$ and $\mathbb{P}[\Phi' \mid \Psi'] = \sum_{\Phi} \mu(\Phi, \Phi')$. Calculating $\Delta(e \,|\, \Psi')$ and $\Delta(e \,|\, \Psi)$ using μ, we see that for any (Φ, Φ') in the support of μ,

$$\begin{aligned} f(\mathrm{dom}(\Psi') \cup \{e\}, \Phi') - f(\mathrm{dom}(\Psi'), \Phi') &= \hat{f}(\Psi' \cup \{(e, \Phi'(e))\}) - \hat{f}(\Psi')) \\ &\leq \hat{f}(\Psi \cup \{(e, \Phi(e))\}) - \hat{f}(\Psi)) \\ &= f(\mathrm{dom}(\Psi) \cup \{e\}, \Phi) - f(\mathrm{dom}(\Psi), \Phi) \end{aligned}$$

from the submodularity of $\hat{f}$. Hence

$$\begin{aligned} \Delta(e \,|\, \Psi') &= \textstyle\sum_{(\Phi, \Phi')} \mu(\Phi, \Phi') \left(f(\mathrm{dom}(\Psi') \cup \{e\}, \Phi') - f(\mathrm{dom}(\Psi'), \Phi') \right) \\ &\leq \textstyle\sum_{(\Phi, \Phi')} \mu(\Phi, \Phi') \left(f(\mathrm{dom}(\Psi) \cup \{e\}, \Phi) - f(\mathrm{dom}(\Psi), \Phi) \right) \qquad = \quad \Delta(e \,|\, \Psi) \end{aligned}$$

which completes the proof. ∎

7. Application: Stochastic Submodular Coverage

Suppose that instead of wishing to adaptively place k unreliable sensors to maximize the utility of the information obtained, as discussed in §6, we have a quota on utility and wish to adaptively place the minimum number of unreliable sensors to achieve this quota. This amounts to a minimum-cost coverage version of the Stochastic Submodular Maximization problem introduced in §6, which we call *Stochastic Submodular Coverage*.

As in §6, in the Stochastic Submodular Coverage problem we suppose there is a function $\hat{f} : 2^{E \times O} \to \mathbb{R}_{\geq 0}$ which quantifies the utility of a set of sensors in arbitrary states. Also, the states of each sensor are independent, so that $\mathbb{P}[\Phi] = \prod_{e \in E} \mathbb{P}[\Phi(e)]$. The goal is to obtain a quota Q of utility at minimum cost. Thus, we define

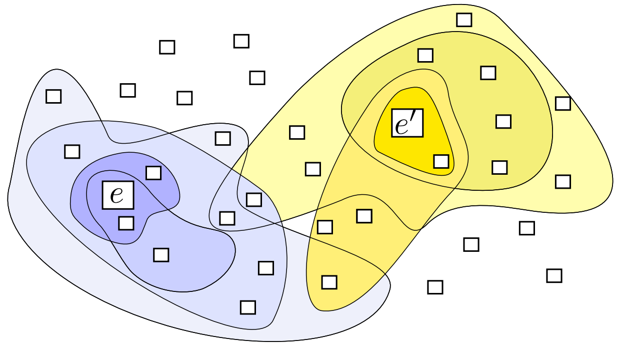


Figure 2: Illustration of part of a Stochastic Set Cover instance. Shown are the supports of two distributions over sets, indexed by items e (marked in blue) and e' (yellow).

our objective as $f(A, \Phi) := \min \left\{Q, \hat{f}(\{(e, \Phi(e)) : e \in A\})\right\}$, and want to find a policy π covering every realization and minimizing $c_{\text{avg}}(\pi) := \mathbb{E}\left[|E(\pi, \Phi)|\right]$. We additionally assume that this quota can always be obtained using sufficiently many sensor placements; formally, this amounts to $f(E, \Phi) = Q$ for all Φ. We obtain the following result, whose proof we defer until the end of this section.

Theorem 14 *Fix a prior such that $\mathbb{P}[\Phi] = \prod_{e \in E} \mathbb{P}[\Phi(e)]$ and let the objective function $\hat{f} : 2^{E \times O} \to \mathbb{R}_{\geq 0}$ be a monotone submodular function. Fix $Q \in \mathbb{R}_{\geq 0}$ such that $f(A, \Phi) := \min\left(Q,\ \hat{f}(\{(e, \Phi(e)) : e \in A\})\right)$ satisfies $f(E, \Phi) = Q$ for all Φ. Let η be any value such that $f(S, \Phi) > Q - \eta$ implies $f(S, \Phi) = Q$ for all S and Φ. Finally, let π be an α-approximate greedy policy for maximizing f, and let π^* be any policy. Then*

$$c_{avg}(\pi) \leq \alpha\, c_{avg}(\pi^*) \left(\ln\left(\frac{Q}{\eta}\right) + 1\right).$$

A Special Case: The Stochastic Set Coverage Problem. The Stochastic Submodular Coverage problem is a generalization of the *Stochastic Set Coverage* problem (Goemans and Vondrák, 2006). In Stochastic Set Coverage the underlying submodular objective $\hat{f}$ is the number of elements covered in some input set system. In other words, there is a ground set U of n elements to be covered, and items E such that each item e is associated with a distribution over subsets of U. When an item is selected, a set is sampled from its distribution, as illustrated in Fig. 2. The problem is to adaptively select items until all elements of U are covered by sampled sets, while minimizing the expected number of items selected. Like us, Goemans and Vondrák also assume that the subsets are sampled independently for each item, and every element of U can be covered in every realization, so that $f(E, \Phi) = |U|$ for all Φ.

Goemans and Vondrák primarily investigated the adaptivity gap of Stochastic Set Coverage, for variants in which items can be repeatedly selected or not, and prove adaptivity gaps of $\Theta(\log n)$ in the former case, and between $\Omega(n)$ and $\mathcal{O}(n^2)$ in the latter. They also provide an n-approximation algorithm. More recently, Liu et al. (2008) considered a special case of Stochastic Set Coverage in which each item may be in one of two states. They were motivated by a streaming database problem, in which a collection of queries sharing common filters must all be evaluated on a stream element. They transform the problem to a Stochastic Set Coverage instance in which (filter, query) pairs are to be covered by filter evaluations; which pairs are covered by a filter depends on the (binary) outcome of evaluating it on the stream element. The resulting instances satisfy the assumption that every element of U can be covered in every realization. They study, among other algorithms, the adaptive greedy algorithm specialized to this setting, and show that if the subsets are sampled independently for each item, so that $\mathbb{P}[\Phi] = \prod_e \mathbb{P}[\Phi(e)]$, then it is an $\mathcal{H}_n := \sum_{x=1}^{n} \frac{1}{x}$ approximation. (Recall $\ln(n) \leq \mathcal{H}_n \leq \ln(n) + 1$ for all $n \geq 1$.) Moreover, Liu et al. report that it empirically outperforms a number of other algorithms in their experiments.

The adaptive submodularity framework allows us to recover Liu et al.'s result, and generalize it to richer item distributions over subsets of U, all as a corollary of Theorem 14. Specifically, we obtain a $(\ln(n)+1)$-approximation for the Stochastic Set Coverage problem, where $n := |U|$, which matches the approximation ratio for the greedy algorithm for classical Set Cover that Stochastic Set Coverage generalizes. Like Liu et al.'s result, our result is tight if NP $\nsubseteq$ DTIME($n^{\mathcal{O}(\log\log n)}$), since it matches Feige's lower bound of $(1-\varepsilon)\ln n$ for the approximability of Set Cover under that assumption (Feige, 1998).

We model the Stochastic Set Coverage problem by letting $\Phi(e) \subseteq U$ indicate the random set sampled from e's distribution. Since the sampled sets are independent we have $\mathbb{P}[\Phi] = \prod_e \mathbb{P}[\Phi(e)]$. For any $A \subseteq E$ let $f(A,\Phi) := |\cup_{e\in A}\Phi(e)|$ be the number of elements of U covered by the sets sampled from items in A. As in the previous work mentioned above, we assume $f(E,\Phi) = n$ for all Φ. Therefore we may set $Q = n$. Since the range of f includes only integers, we may set $\eta = 1$. Applying Theorem 14 then yields the following result.

Corollary 15 *The adaptive greedy algorithm achieves a* $(\ln(n)+1)$*-approximation for Stochastic Set Coverage, where* $n := |U|$ *is the size of the ground set.*

We now provide the proof of Theorem 14.

Proof of Theorem 14: We will ultimately prove Theorem 14 by applying the bound from Theorem 10 for self–certifying instances. The proof mostly consists of justifying this final step. Without loss of generality we may assume $\hat{f}$ is truncated at Q, otherwise we may use $\hat{g}(S) = \min\left\{Q, \hat{f}(S)\right\}$ in lieu of $\hat{f}$. This removes the need to truncate f. Since we established the adaptive submodularity of f in the proof of Theorem 13, and by assumption $f(E,\Phi) = Q$ for all Φ, to apply Theorem 10 we need only show that f is strongly adaptive monotone, and that the instances under consideration are self–certifying.

We begin by showing the strong adaptive monotonicity of f. Fix a partial realization Ψ, an item $e \notin \mathrm{dom}(\Psi)$ and a state o. Let $\Psi' = \Psi \cup \{(e,o)\}$. Then treating Ψ and Ψ' as subsets of $E \times O$, and using the monotonicity of $\hat{f}$, we obtain

$$\mathbb{E}\left[f(\mathrm{dom}(\Psi),\Phi) \mid \Phi \sim \Psi\right] = \hat{f}(\Psi) \leq \hat{f}(\Psi') \leq \mathbb{E}\left[f(\mathrm{dom}(\Psi'),\Phi) \mid \Phi \sim \Psi'\right],$$

which is equivalent to the strong adaptive monotonicity condition.

Next we prove that these instances are self–certifying. Consider any Ψ and Φ, Φ' consistent with Ψ. Then

$$f(\mathrm{dom}(\Psi),\Phi) = \hat{f}(\Psi) = f(\mathrm{dom}(\Psi),\Phi').$$

Since $f(E,\Phi) = f(E,\Phi') = Q$ by assumption, it follows that $f(\mathrm{dom}(\Psi),\Phi) = f(E,\Phi)$ iff $f(\mathrm{dom}(\Psi),\Phi') = f(E,\Phi')$, so the instance is self–certifying.

We have shown that f and $\mathbb{P}[\Phi]$ satisfy the assumptions of Theorem 10 on this self–certifying instance. Hence we may apply it to obtain the claimed approximation guarantee. ■

8. Application: Adaptive Viral Marketing

For our next application, consider the following scenario. Suppose we would like to generate demand for a genuinely novel product. Potential customers do not realize how valuable the new product will be to them, and conventional advertisements are failing to convince them to try it. In this case, we may try to spur demand by offering a special promotional deal to a select few people, and hope that demand builds virally, propagating through the social network as people recommend the product to their friends and associates. Supposing we know something about the structure of the social networks people inhabit, and how ideas, innovation, and new product adoption diffuse through them, this begs the question: to which initial set of people should we offer the promotional deal, in order to spur maximum demand for our product?

This, broadly, is the viral marketing problem. The same problem arises in the context of spreading technological, cultural, and intellectual innovations, broadly construed. In the interest of unified terminology

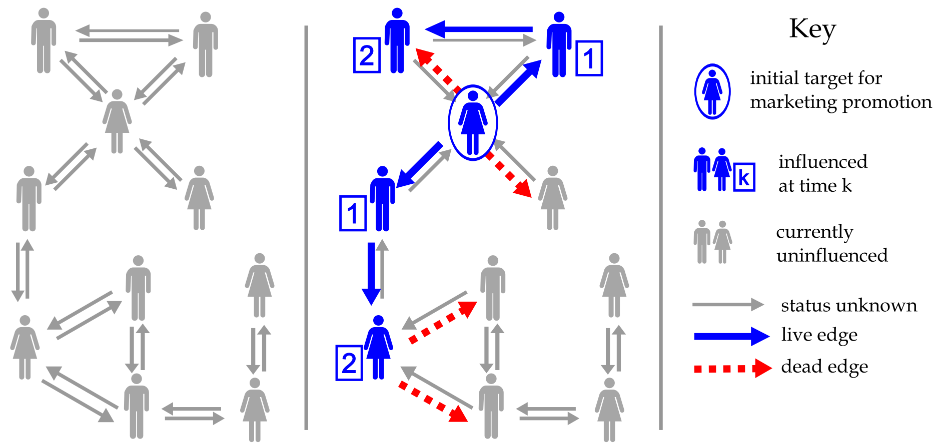


Figure 3: Illustration of the Adaptive Viral Marketing problem. Left: the underlying social network. Middle: the people influenced and the observations obtained after one person is selected.

we follow Kempe et al. (2003) and talk of spreading *influence* through the social network, where we say people are *active* if they have adopted the idea or innovation in question, and *inactive* otherwise, and that a *influences* b if a convinces b to adopt the idea or innovation in question.

There are many ways to model the diffusion dynamics governing the spread of influence in a social network. We consider a basic and well-studied model, the *independent cascade model*, described in detail below. For this model Kempe et al. obtain a very interesting result; they show that the eventual spread of the influence f (i.e., the ultimate number of customers that demand the product) is a monotone submodular function of the seed set S of people initially selected. This, in conjunction with the results of Nemhauser et al. (1978) implies that the greedy algorithm obtains at least $\left(1 - \frac{1}{e}\right)$ of the value of the best feasible seed set of size at most k, i.e., $\arg\max_{S:|S|\leq k} f(S)$, where we interpret k as the budget for the promotional campaign. Though Kempe et al. consider only the maximum coverage version of the viral marketing problem, their result in conjunction with that of Wolsey (1982) also implies that the greedy algorithm will obtain a quota Q of value at a cost of at most $\ln(Q) + 1$ times the cost of the optimal set $\arg\min_S \{c(S) : f(S) \geq Q\}$ if f takes on only integral values.

Adaptive Viral Marketing. The viral marketing problem has a natural adaptive analog. Instead of selecting a fixed set of people in advance, we may select a person to offer the promotion to, make some observations about the resulting spread of demand for our product, and repeat. See Fig. 3 for an illustration. In §8.1, we use the idea of adaptive submodularity to obtain results analogous to those of Kempe et al. (2003) in the adaptive setting. Specifically, we show that the greedy policy obtains at least $\left(1 - \frac{1}{e}\right)$ of the value of the best *policy*. Moreover, we extend this result by achieving that guarantee not only for the case where our reward is simply the number of influenced people, but also for any (nonnegative) monotone submodular function of the *set* of people influenced. In §8.2 we consider the minimum cost cover objective, and show that the greedy policy obtains a logarithmic approximation for it.

Independent Cascade Model. In this model, the social network is a directed graph $G = (V, A)$ where each vertex in V is a person, and each edge $(u, v) \in A$ has an associated binary random variable X_{uv} indicating if u will influence v. That is, $X_{uv} = 1$ if u will influence v once it has been influenced, and $X_{uv} = 0$ otherwise. The random variables X_{uv} are independent, and have known means $p_{uv} := \mathbb{E}[X_{uv}]$. We will call an edge (u, v) with $X_{uv} = 1$ a *live edge* and an edge with $X_{uv} = 0$ a *dead edge*. When a node u is activated, the edges X_{uv} to each neighbor v of u are sampled, and v is activated if (u, v) is live. Influence can then spread from u's neighbors to their neighbors, and so on, according to the same process. Once active, nodes remain active

throughout the process, however Kempe et al. (2003) show that this assumption is without loss of generality, and can be removed.

The Feedback Model. In the Adaptive Viral Marketing problem under the independent cascades model, the items correspond to people we can activate by offering them the promotional deal. How we define the states $\Phi(u)$ depends on what information we obtain as a result of activating u. Given the nature of the diffusion process, activating u can have wide-ranging effects, so the state $\Phi(u)$ has more to do with the state of the social network on the whole than with u in particular. Specifically, we model $\Phi(u)$ as a function $\Phi_u : A \to \{0, 1, ?\}$, where $\Phi_u((u, v)) = 0$ means that activating u has revealed that (u, v) is dead, $\Phi_u((u, v)) = 1$ means that activating u has revealed that (u, v) is live, and $\Phi_u((u, v)) = ?$ means that activating u has not revealed the status of (u, v) (i.e., the value of X_{uv}). We require each realization to be *consistent* and *complete*. Consistency means that no edge should be declared both live and dead by any two states. That is, for all $u, v \in V$ and $a \in A$, $(\Phi_u(a), \Phi_v(a)) \notin \{(0, 1), (1, 0)\}$. Completeness means that the status of each edge is revealed by some activation. That is, for all $a \in A$ there exists $u \in V$ such that $\Phi_u(a) \in \{0, 1\}$. A consistent and complete realization thus encodes X_{uv} for each edge (u, v). Let $A(\Phi)$ denote the live edges as encoded by Φ. There are several candidates for which edge sets we are allowed to observe when activating a node u. Here we consider what we call the *Full-Adoption Feedback Model*: After activating u we get to see the status (live or dead) of all edges exiting v, for all nodes v reachable from u via live edges (i.e., reachable from u in $(V, A(\Phi))$, where Φ is the true realization. We illustrate the full-adoption feedback model in Fig. 3.

The Objective Function. In the simplest case, the reward for influencing a set $U \subseteq V$ of nodes is $\hat{f}(U) := |U|$. Kempe et al. (2003) obtain an $\left(1 - \frac{1}{e}\right)$-approximation for the slightly more general case in which each node u has a weight w_u indicating its importance, and the reward is $\hat{f}(U) := \sum_{u \in U} w_u$. We generalize this result further, to include arbitrary nonnegative monotone submodular reward functions $\hat{f}$. This allows us, for example, to encode a value associated with the *diversity* of the set of nodes influenced, such as the notion that it is better to achieve 20% market penetration in five different (equally important) demographic segments than 100% market penetration in one and 0% in the others.

8.1 The Maximum Coverage Objective

We are now ready to formally state our result for the maximum coverage objective.

Theorem 16 *The greedy policy π^{greedy} obtains at least $\left(1 - \frac{1}{e}\right)$ of the value of the best* policy *for the Adaptive Viral Marketing problem with arbitrary monotone submodular reward functions, in the independent cascade and full-adoption feedback models discussed above. That is, if $\sigma(S, \Phi)$ is the set of all activated nodes when S is the seed set of activated nodes and Φ is the realization, $\hat{f} : 2^V \to \mathbb{R}_{\geq 0}$ is an arbitrary monotone submodular function indicating the reward for influencing a set, and the objective function is $f(S, \Phi) := \hat{f}(\sigma(S, \Phi))$, then for all policies π and all $k \in \mathbb{N}$ we have*

$$f_{avg}(\pi^{greedy}_{[k]}) \geq \left(1 - \frac{1}{e}\right) f_{avg}(\pi_{[k]}).$$

More generally, if π is an α-approximate greedy policy then for all $\ell \in \mathbb{N}$, $f_{avg}(\pi_{[\ell]}) \geq \left(1 - e^{-\ell/\alpha k}\right) f_{avg}(\pi^*_{[k]})$.

Proof Adaptive monotonicity follows immediately from the fact that $f(\cdot, \Phi)$ is monotonic for each Φ. It thus suffices to prove that f is adaptive submodular with respect to the probability distribution on realizations $\mathbb{P}[\Phi]$, because then we can invoke Theorem 5 to complete the proof.

We will say we have *observed* an edge (u, v) if we know its status, i.e., if it is live or dead. Fix any Ψ, Ψ' such that $\Psi \subseteq \Psi'$ and any $v \notin \mathrm{dom}(\Psi')$. We must show $\Delta(v \,|\, \Psi') \leq \Delta(v \,|\, \Psi)$. To prove this rigorously, we define a coupled distribution μ over pairs of realizations $\Phi \sim \Psi$ and $\Phi' \sim \Psi'$. Note that given the feedback model, the realization Φ is a function of the random variables $\{X_{uw} : (u, w) \in A\}$ indicating the status of

each edge. For conciseness we use the notation $\mathbf{X} = \{X_{uw} : (u, w) \in A\}$. We define μ implicitly in terms of a joint distribution $\hat{\mu}$ on $\mathbf{X} \times \mathbf{X}'$, where $\Phi = \Phi(\mathbf{X})$ and $\Phi' = \Phi'(\mathbf{X}')$ are the realizations induced by the two distinct sets of random edge statuses, respectively. Hence $\mu(\Phi(\mathbf{X}), \Phi(\mathbf{X}')) = \hat{\mu}(\mathbf{X}, \mathbf{X}')$. Next, let us say a partial realization Ψ observes an edge e if some $w \in \mathrm{dom}(\Psi)$ has revealed its status as being live or dead. For edges (u, w) observed by Ψ, the random variable X_{uw} is deterministically set to the status observed by Ψ. Similarly, for edges (u, w) observed by Ψ', the random variable X'_{uw} is deterministically set to the status observed by Ψ'. Note that since $\Psi \subseteq \Psi'$, the state of all edges which are observed by Ψ are the same in Φ and Φ'. All $(\mathbf{X}, \mathbf{X}') \in \mathrm{support}(\hat{\mu})$ have these properties. Additionally, we will construct $\hat{\mu}$ so that the status of all edges which are unobserved by both Ψ' and Ψ are the same in $\mathbf{X}$ and $\mathbf{X}'$, meaning $X_{uw} = X'_{uw}$ for all such edges (u, w), or else $\hat{\mu}(\mathbf{X}, \mathbf{X}') = 0$.

The above constraints leave us with the following degrees of freedom: we may select X_{uw} for all $(u, w) \in A$ which are unobserved by Ψ. We select them independently, such that $\mathbb{E}[X_{uw}] = p_{uw}$ as with $\mathbb{P}[\Phi]$. Hence for all $(\mathbf{X}, \mathbf{X}')$ satisfying the above constraints,

$$\hat{\mu}(\mathbf{X}, \mathbf{X}') = \prod_{(u,w) \text{ unobserved by } \Psi} p_{uw}^{X_{uw}} (1 - p_{uw})^{1 - X_{uw}},$$

and otherwise $\hat{\mu}(\mathbf{X}, \mathbf{X}') = 0$. Note that $\mathbb{P}[\Phi \mid \Psi] = \sum_{\Phi'} \mu(\Phi, \Phi')$ and $\mathbb{P}[\Phi' \mid \Psi'] = \sum_{\Phi} \mu(\Phi, \Phi')$. We next claim that for all $(\Phi, \Phi') \in \mathrm{support}(\mu)$

$$f(\mathrm{dom}(\Psi') \cup \{v\}, \Phi') - f(\mathrm{dom}(\Psi'), \Phi') \quad \leq \quad f(\mathrm{dom}(\Psi) \cup \{v\}, \Phi) - f(\mathrm{dom}(\Psi), \Phi) \tag{17}$$

Recall $f(S, \Phi) := \hat{f}(\sigma(S, \Phi))$, where $\sigma(S, \Phi)$ is the set of all activated nodes when S is the seed set of activated nodes and Φ is the realization. Let $B = \sigma(\mathrm{dom}(\Psi), \Phi)$ and $C = \sigma(\mathrm{dom}(\Psi) \cup \{v\}, \Phi)$ denote the active nodes before and after selecting v after $\mathrm{dom}(\Psi)$ under realizations Φ, and similarly define B' and C' with respect to Ψ' and Φ'. Let $D := C \setminus B$, $D' := C' \setminus B'$. Then Eq. (17) is equivalent to $\hat{f}(B' \cup D') - \hat{f}(B') \leq \hat{f}(B \cup D) - \hat{f}(B)$. By the submodularity of $\hat{f}$, it suffices to show that $B \subseteq B'$ and $D' \subseteq D$ to prove the above inequality, which we will now do.

We start by proving $B \subseteq B'$. Fix $w \in B$. Then there exists a path from some $u \in \mathrm{dom}(\Psi)$ to w in $(V, A(\Phi))$. Moreover, every edge in this path is not only live but also observed to be live, by definition of the feedback model. Since $(\Phi, \Phi') \in \mathrm{support}(\mu)$, this implies that every edge in this path is also live under Φ', as edges observed by Ψ must have the same status under both Φ and Φ'. It follows that there is a path from u to w in $(V, A(\Phi'))$. Since u is clearly also in $\mathrm{dom}(\Psi')$, we conclude $w \in B'$, hence $B \subseteq B'$.

Next we show $D' \subseteq D$. Fix some $w \in D'$ and suppose by way of contradiction that $w \notin D$. Hence there exists a path P from v to w in $(V, A(\Phi'))$ but no such path exists in $(V, A(\Phi))$. The edges of P are all live under Φ', and at least one must be dead under Φ. Let (u, u') be such an edge in P. Because the status of this edge differs in Φ and Φ', and $(\Phi, \Phi') \in \mathrm{support}(\mu)$, it must be that (u, u') is observed by Ψ' but not observed by Ψ. Because it is observed by Ψ', in our feedback model it must be that u is active after $\mathrm{dom}(\Psi')$ is selected, i.e., $u \in B'$. However, this implies that all nodes reachable from u via edges in P are also active after $\mathrm{dom}(\Psi')$ is selected, since all the edges in P are live. Hence all such nodes, including w, are in B'. Since D' and B' are disjoint, this implies $w \notin D'$, a contradiction.

Having proved Eq. (17), we now proceed to use it to show $\Delta(v \,|\, \Psi') \leq \Delta(v \,|\, \Psi)$ as in §6.

$$\begin{aligned} \Delta(v \,|\, \Psi') &= \textstyle\sum_{(\Phi, \Phi')} \mu(\Phi, \Phi') \left(f(\mathrm{dom}(\Psi') \cup \{v\}, \Phi') - f(\mathrm{dom}(\Psi'), \Phi') \right) \\ &\leq \textstyle\sum_{(\Phi, \Phi')} \mu(\Phi, \Phi') \left(f(\mathrm{dom}(\Psi) \cup \{v\}, \Phi) - f(\mathrm{dom}(\Psi), \Phi) \right) \quad = \quad \Delta(v \,|\, \Psi) \end{aligned}$$

which completes the proof. ■

Comparison with Stochastic Submodular Maximization. It is worth contrasting the Adaptive Viral Marketing problem with the Stochastic Submodular Maximization problem of §6. In the latter problem, we can

think of the items as being random independently distributed sets. In Adaptive Viral Marketing by contrast, the random sets (of nodes influenced when a fixed node is selected) depend on the random status of the edges, and hence may be correlated through them. Nevertheless, we can obtain the same $\left(1 - \frac{1}{e}\right)$ approximation factor for both problems.

A Comment on the Myopic Feedback Model. In the conference version of this article (Golovin and Krause, 2010), we considered an alternate feedback model called the *myopic feedback* model, in which after activating v we see the status of all edges exiting v in the social network, i.e., $\partial_+(u) := \{(u, v) : v \in V\} \cap A$. We claimed that the objective f as defined previously is adaptive submodular in the independent cascade model with myopic feedback, and hence the greedy policy obtains a $(1 - \frac{1}{e})$ approximation for it. We hereby retract this claim, and furthermore give a counterexample demonstrating that f is not adaptive submodular under myopic feedback.

Consider a graph $G = (V, E)$ with vertices $V := \{u, v, w\}$, and edges $E := \{(u, v), (v, w)\}$. The edge parameters are $p_{uv} = 1$ and $p_{vw} = 1 - \epsilon$. Let $\hat{f}(U) = |U|$ and construct f from $\hat{f}$ accordingly. We let $\Psi = \{(u, \Phi_u)\}$, where $\Phi_u((u, v)) = 1$ and $\Phi_u((v, w)) = ?$. Let $\Psi' = \{(u, \Phi_u), (v, \Phi_v)\}$ where $\Phi_v((v, w)) = 0$. Clearly, $\Psi \subset \Psi'$. Note $\Delta(w \,|\, \Psi) = \epsilon$, since the marginal benefit of w over $\mathrm{dom}(\Psi)$ is one if (v, w) is dead, and zero if it is live, and the former occurs with probability ϵ. In contrast, $\Delta(w \,|\, \Psi') = 1$, since Ψ' contains the observation that (v, w) is dead. Hence $\Delta(w \,|\, \Psi) < \Delta(w \,|\, \Psi')$, which violates adaptive submodularity. However, we conjecture that the greedy policy still obtains a constant factor approximation even in the myopic feedback model.

8.2 The Minimum Cost Cover Objective

For the minimum cost cover objective, we have an instance of Adaptive Stochastic Minimum Cost Cover, in which we are given a quota $Q \leq \hat{f}(V)$ and we must adaptively select nodes to activate until the set of all active nodes S satisfies $\hat{f}(S) \geq Q$. We obtain the following result.

Theorem 17 *Fix a monotone submodular function $\hat{f} : 2^V \to \mathbb{R}_{\geq 0}$ indicating the reward for influencing a set, and a quota $Q \leq \hat{f}(V)$. Suppose the objective function is $f(S, \Phi) := \min\left\{Q, \hat{f}(\sigma(S, \Phi))\right\}$, where $\sigma(S, \Phi)$ is the set of all activated nodes when S is the seed set of activated nodes and Φ is the realization. Let η be any value such that $\hat{f}(S) > Q - \eta$ implies $\hat{f}(S) \geq Q$ for all S. Then any α-approximate greedy policy π on average costs at most $\alpha\left(\ln\left(\frac{Q}{\eta}\right) + 1\right)$ times the average cost of the best* policy *obtaining Q reward for the Adaptive Viral Marketing problem in the independent cascade model with full-adoption feedback as described above. That is, $c_{avg}(\pi) \leq \alpha\left(\ln\left(\frac{Q}{\eta}\right) + 1\right) c_{avg}(\pi^*)$ for any π^* that covers every realization.*

Proof We prove Theorem 17 by recourse to Theorem 10. We have already established that f is adaptive submodular, in the proof of Theorem 16. It remains to show that f is strongly adaptive monotone, that these instances are self–certifying, and that Q and η equal the corresponding terms in the statement of Theorem 10.

We start with strong adaptive monotonicity. Fix Ψ, $e \notin \mathrm{dom}(\Psi)$, and $o \in O$. We must show

$$\mathbb{E}\left[f(\mathrm{dom}(\Psi), \Phi) \;\mid\; \Phi \sim \Psi\right] \leq \mathbb{E}\left[f(\mathrm{dom}(\Psi) \cup \{e\}, \Phi) \;\mid\; \Phi \sim \Psi, \Phi(e) = o\right]. \tag{18}$$

Let $V^+(\Psi)$ denote the active nodes after selecting $\mathrm{dom}(\Psi)$ and observing Ψ. By definition of the full adoption feedback model, $V^+(\Psi)$ consists of precisely those nodes v for which there exists a path P_{uv} from some $u \in \mathrm{dom}(\Psi)$ to v via exclusively live edges. The edges whose status we observe consist of all edges exiting nodes in $V^+(\Psi)$. It follows that every path from any $u \in V^+(\Psi)$ to any $v \in V \setminus V^+(\Psi)$ contains at least one edge which is observed by Ψ to be dead. Hence, in every $\Phi \sim \Psi$, the set of nodes activated by selecting $\mathrm{dom}(\Psi)$ is the same. Therefore $\mathbb{E}\left[f(\mathrm{dom}(\Psi), \Phi) \;\mid\; \Phi \sim \Psi\right] = \hat{f}(V^+(\Psi))$. Similarly, if we define $\Psi' := \Psi \cup \{(e, o)\}$, then $\mathbb{E}\left[f(\mathrm{dom}(\Psi) \cup \{e\}, \Phi) \;\mid\; \Phi \sim \Psi, \Phi(e) = o\right] = \hat{f}(V^+(\Psi'))$. Note that once activated, nodes never become inactive. Hence, $\Psi \subseteq \Psi'$ implies $V^+(\Psi) \subseteq V^+(\Psi')$. Since $\hat{f}$ is monotone by assumption, this means $\hat{f}(V^+(\Psi)) \leq \hat{f}(V^+(\Psi'))$ which implies Eq. (18) and strong adaptive monotonicity.

Next we establish that these instances are self–certifying. Note that for every Φ we have $f(V, \Phi) = \min\left\{Q, \hat{f}(V)\right\} = Q$. From our earlier remarks, we know that $f(\mathrm{dom}(\Psi), \Phi) = \hat{f}(V^+(\Psi))$ for every $\Phi \sim \Psi$. Hence for all Ψ and Φ, Φ' consistent with Ψ, we have $f(\mathrm{dom}(\Psi), \Phi) = f(\mathrm{dom}(\Psi), \Phi')$ and so $f(\mathrm{dom}(\Psi), \Phi) = Q$ if and only if $f(\mathrm{dom}(\Psi), \Phi') = Q$, which proves that the instance is self–certifying.

Finally we show that Q and η equal the corresponding terms in the statement of Theorem 10. As noted earlier, $f(V, \Phi) = Q$ for all Φ. We defined η as some value such that $\hat{f}(S) > Q - \eta$ implies $\hat{f}(S) \geq Q$ for all S. Since $\mathrm{range}(f) = \left\{\min\left\{Q, \hat{f}(S)\right\} : S \subseteq V\right\}$, it follows that we cannot have $f(S, \Phi) \in (Q - \eta, Q)$ for any S and Φ, so that η satisfies the requirements of the corresponding term in Theorem 10. Hence we may apply Theorem 10 on this self–certifying instance with Q and η to obtain the claimed result. ■

9. Application: Active Learning

Obtaining labeled data to train a classifier is typically expensive, as it often involves asking an expert. In active learning (*c.f.*, Cohn et al. (1996), McCallum and Nigam (1998)), the key idea is that some labels are more informative than others: labeling a few unlabeled examples can imply the labels of many other unlabeled examples, and thus the cost of obtaining the labels from an expert can be avoided. As is standard, we assume that we are given a set of hypotheses H, and a set of unlabeled data points X where each $x \in X$ is independently drawn from some distribution $\mathcal{D}$. Let L be the set of possible labels. Classical learning theory yields *probably approximately correct* (PAC) bounds, bounding the number n of examples drawn i.i.d. from $\mathcal{D}$ needed to output a hypothesis h that will have expected error at most ε with probability at least $1 - \delta$, for some fixed $\varepsilon, \delta > 0$. That is, if h^* is the target hypothesis (with zero error), and $\mathrm{error}(h) := \mathbb{P}_{x \sim \mathcal{D}}\left[h(x) \neq h^*(x)\right]$ is the error of h, we require $\mathbb{P}\left[\mathrm{error}(h) \leq \varepsilon\right] \geq 1 - \delta$. The latter probability is taken with respect to $\mathcal{D}(X)$; the learned hypothesis h and thus $\mathrm{error}(h)$ depend on it.

A key challenge in active learning is to avoid bias: actively selected examples are no longer i.i.d., and thus sample complexity bounds for passive learning no longer apply. If one is not careful, active learning may require more samples than passive learning to achieve the same generalization error. One natural approach to active learning that is guaranteed to perform at least as well as passive learning is *pool-based active learning* (McCallum and Nigam, 1998): The idea is to draw n *unlabeled* examples i.i.d. However, instead of obtaining all labels, labels are adaptively requested until the labels of all unlabeled examples are implied by the obtained labels. Now we have obtained n labeled examples drawn i.i.d., and classical PAC bounds still apply. The key question is how to request the labels for the pool to infer the remaining labels as quickly as possible.

In the case of binary labels $L = \{-1, 1\}$, various authors have considered greedy policies which generalize binary search (Garey and Graham, 1974; Loveland, 1985; Arkin et al., 1993; Kosaraju et al., 1999; Dasgupta, 2004; Guillory and Bilmes, 2009; Nowak, 2009). The simplest of these, called *generalized binary search* (GBS) or the *splitting algorithm*, works as follows. Define the *version space* V to be the set of hypotheses consistent with the observed labels (here we assume that there is no label noise). In the worst-case setting, GBS selects a query $x \in X$ that minimizes $\left|\sum_{h \in V} h(x)\right|$. In the Bayesian setting we assume we are given a prior p_H over hypotheses; in this case GBS selects a query $x \in X$ that minimizes $\left|\sum_{h \in V} p_H(h) \cdot h(x)\right|$. Intuitively these policies myopically attempt to shrink a measure of the version space (i.e., the cardinality or the probability mass) as quickly as possible. The former provides an $\mathcal{O}(\log |H|)$-approximation for the worst-case number of queries (Arkin et al., 1993), and the latter provides an $\mathcal{O}(\log \frac{1}{\min_h p_H(h)})$-approximation for the expected number of queries (Kosaraju et al., 1999; Dasgupta, 2004) and a natural generalization of GBS obtains the same guarantees with a larger set of labels (Guillory and Bilmes, 2009). Kosaraju et al. also prove that running GBS on a modified prior $p'_H(h) \propto \max\left\{p_H(h), 1/|H|^2 \log |H|\right\}$ is sufficient to obtain an $\mathcal{O}(\log |H|)$-approximation.

Viewed from this perspective of the previous sections, shrinking the version space amounts to "covering" all false hypotheses with stochastic sets (i.e., queries), where query x covers all hypotheses that disagree with the target hypothesis h^* at x. That is, x covers $\{h : h(x) \neq h^*(x)\}$. As in §8, these sets may be correlated in

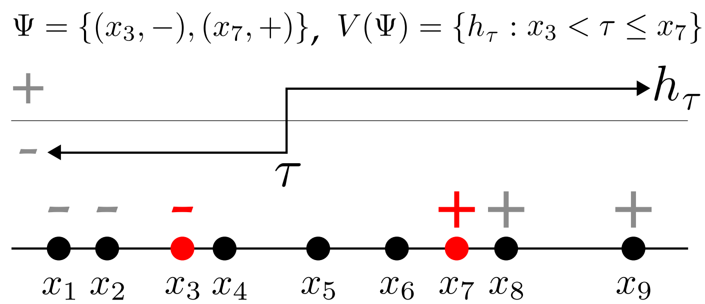


Figure 4: Illustration of the Active Learning problem, in the simple special case of one-dimensional data and binary threshold hypotheses $H = \{h_\tau : \tau \in \mathbb{R}\}$, where $h_\tau(x) = 1$ if $x \geq \tau$ and 0 otherwise.

complex ways determined by the set of possible hypotheses. As we will show, the reduction in version space mass is adaptive submodular, and this allows us to obtain a new analysis of GBS using adaptive submodularity, which is arguably more amenable to extensions and generalizations than previous analyses. Our new analysis further allows us to improve on the previous best bound on the approximation factor of GBS (Dasgupta, 2004) from $4 \ln\left(\frac{1}{\min_h p_H(h)}\right)$ to $\ln\left(\frac{1}{\min_h p_H(h)}\right) + 1$. We also show that when we apply GBS to a modified prior distribution, the approximation factor is improved to $\mathcal{O}(\ln |H|)$. This result matches a lower bound of $\Omega(\ln |H|)$ of Chakaravarthy et al. (2007) up to constant factors.

Theorem 18 *In the Bayesian setting in which there is a prior p_H on a finite set of hypotheses H, the generalized binary search algorithm makes $\mathsf{OPT} \cdot \left(\ln\left(\frac{1}{\min_h p_H(h)}\right) + 1\right)$ queries in expectation to identify a hypothesis drawn from p_H, where OPT is the minimum expected number of queries made by any policy. If $\min_h p_H(h)$ is sufficiently small, running the algorithm on a modified prior $p'_H(h) \propto \max\left\{p_H(h), 1/|H|^2\right\}$ improves the approximation factor to $\mathcal{O}(\ln |H|)$.*

Proof We first address the important special case of a uniform prior over hypotheses, i.e., $p_H(h) = 1/|H|$ for all $h \in H$, and then we reduce the case with a general prior to a uniform prior. We wish to appeal to Theorem 10, so we convert the problem into an Adaptive Stochastic Min Cost Cover problem. Define a realization Φ_h for each hypothesis $h \in H$. The ground set is $E = X$, and the outcomes are binary; we define $O = \{-1, 1\}$ instead of using $\{0, 1\}$ to be consistent with our earlier exposition. For all $h \in H$ we set $\Phi_h \equiv h$, meaning $\Phi_h(x) = h(x)$ for all $x \in X$. To define the objective function, we first need some notation. Given observed labels $\Psi \subset X \times O$, let $V(\Psi)$ denote the version space, i.e., the set of hypotheses for which $h(x) = \Psi(x)$ for all $x \in \mathrm{dom}(\Psi)$. See Fig. 4 for an illustration of an active learning problem in the case of indicator hypotheses. For a set of hypotheses V, let $p_H(V) := \sum_{h \in V} p_H(h)$ denote their total prior probability. Finally, let $\Psi(S, h) = \{(x, h(x)) : x \in S\}$ be the function with domain S that agrees with h on S. We define the objective function by

$$f(S, \Phi_h) := 1 - p_H(V(\Psi(S, h))) = p_H(\{h' : \exists x \in S, h'(x) \neq h(x)\})$$

and use $\mathbb{P}[\Phi_h] = p_H(h) = 1/|H|$ for all h. Let π^* be an optimal policy for this Adaptive Stochastic Min Cost Cover instance. Note that there is an exact correspondence between policies for the original problem of finding the target hypothesis and our problem of covering the true realization; identifying h^* as the target hypothesis corresponds to covering Φ_{h^*}. Hence $c_{\text{avg}}(\pi^*) = \mathsf{OPT}$. Note that because we have assumed a uniform prior over hypotheses, we have $f(X, \Phi_h) = 1 - 1/|H|$ for all h. Also, maximizing the conditional expected reward forces the policy to identify h^* and hence Φ_{h^*}, and this ensures that these instances are self–certifying. More formally, these instances are self–certifying because for any Φ_h and Ψ such that $\Phi_h \sim \Psi$, we have that $f(\mathrm{dom}(\Psi), \Phi_h) = f(X, \Phi_h)$ implies $V(\Psi) = \{h\}$. This in turn means that Φ_h is the *only* realization

consistent with Ψ, which trivially implies that any realization $\Phi' \sim \Psi$ also has $f(\mathrm{dom}(\Psi), \Phi') = f(X, \Phi')$; hence the instance is self–certifying.

We next argue that f is adaptive submodular and strongly adaptive monotone with respect to $\mathbb{P}[\Phi]$. Each query x eliminates some subset of hypotheses, and as more queries are performed, the subset of hypotheses eliminated by x cannot grow. More formally, consider the expected marginal contribution of x under two partial realizations Ψ, Ψ' where Ψ is a subrealization of Ψ' (i.e., $\Psi \subset \Psi'$), and $x \notin \mathrm{dom}(\Psi')$. Let $\Psi[x/o]$ be the partial realization with domain $\mathrm{dom}(\Psi) \cup \{x\}$ that agrees with Ψ on its domain, and maps x to o. For each $o \in O$, let $a_o := p_H(V(\Psi[x/o]))$, $b_o := p_H(V(\Psi'[x/o]))$. Since a hypothesis eliminated from the version space cannot later appear in the version space, we have $a_o \geq b_o$ for all o. Next, note the expected reduction in version space mass (and hence the expected marginal contribution) due to selecting x given partial realization Ψ is

$$\Delta(x \,|\, \Psi) = \sum_{o \in O} a_o \cdot \mathbb{P}[\Phi(x) \neq o \;\mid\; \Phi \sim \Psi] = \sum_o a_o \left(\frac{\sum_{o' \neq o} a_{o'}}{\sum_{o'} a_{o'}} \right) = \frac{\sum_{o \neq o'} a_o a_{o'}}{\sum_{o'} a_{o'}} \tag{19}$$

The corresponding quantity for Ψ' has b_o substituted for a_o in Eq. (19), for each o. To prove adaptive submodularity we must show $\Delta(x \,|\, \Psi) \geq \Delta(x \,|\, \Psi')$ and to do so it suffices to show that $\partial\phi/\partial z_o \geq 0$ for each o and $\vec{z} \in \left\{\vec{c} \in [0,1]^O : \sum_o c_o > 0\right\}$, where $\phi(\vec{z}) := \left(\sum_{o \neq o'} z_o z_{o'}\right) / \left(\sum_{o'} z_{o'}\right)$ has the same functional form as the expression for $\Delta(x \,|\, \Psi)$ in Eq. (19). This is because $\partial\phi/\partial z_o \geq 0$ for each o implies that growing the version space in any manner cannot decrease the expected marginal benefit of query x, and hence shrinking it in any manner cannot increase the expected marginal benefit of x. It is indeed the case that $\partial\phi/\partial z_o \geq 0$ for each o. More specifically, it holds that

$$\frac{\partial \phi}{\partial z_a} = \frac{\sum_{b \neq a} z_b^2 + \sum_{(b,c): b \neq c, b \neq a, c \neq a} z_b z_c}{\left(\sum_b z_b\right)^2} \geq 0,$$

which can be derived through elementary calculus.

Demonstrating strong adaptive monotonicity amounts to proving that adding labels cannot grow the version space, which is clear in our model. Hence we can apply Theorem 10 to this self–certifying instance with maximum reward threshold $Q = 1 - 1/|H|$, and minimum gap $\eta = 1/|H|$, to obtain an upper bound of $\mathsf{OPT}\,(\ln(|H| - 1) + 1)$ on the number of queries made by the generalized binary search algorithm (which corresponds exactly to the greedy policy for Adaptive Stochastic Min Cost Cover) under the assumption of a uniform prior over H.

Now consider general priors over H. We construct the Adaptive Stochastic Min Cost Cover instance as before, only we change the objective function to

$$f(S, \Phi_h) \;:=\; 1 - p_H(V(\Psi(S, h))) + p_H(h).$$

The modified objective is still adaptive submodular, because $(S, \Phi_h) \mapsto p_H(h)$ is clearly so, and because adaptive submodularity is defined via linear inequalities it is preserved under taking nonnegative linear combinations. Note that $f(X, \Phi_h) = 1$ for all Φ_h. Showing f is strongly adaptive monotone requires slightly more work than before. Fix $\Psi, x \notin \mathrm{dom}(\Psi)$, and $o \in O$. We must show $\mathbb{E}[f(\mathrm{dom}(\Psi), \Phi) \;\mid\; \Phi \sim \Psi] \leq \mathbb{E}[f(\mathrm{dom}(\Psi) \cup \{x\}, \Phi) \;\mid\; \Phi \sim \Psi, \Phi(x) = o]$. Plugging in the definition of f, the inequality we wish to prove may be simplified to

$$\mathbb{E}[p_H(h) \;\mid\; \Phi_h \sim \Psi] - \mathbb{E}[p_H(h) \;\mid\; \Phi_h \sim \Psi[x/o]] \;\leq\; p_H(V(\Psi)) - p_H(V(\Psi[x/o])). \tag{20}$$

where the expectations are taken over Φ_h. Let $V_{\text{elim}} := V(\Psi) - V(\Psi[x/o])$ be the set of hypotheses eliminated from the version space by the observation $h(x) = o$. Rewriting Eq. (20), we get

$$\sum_{h \in V(\Psi)} \frac{p_H(h)^2}{p_H(V(\Psi))} - \sum_{h \in V(\Psi[x/o])} \frac{p_H(h)^2}{p_H(V(\Psi[x/o]))} \;\leq\; p_H(V_{\text{elim}}). \tag{21}$$

Let LHS_{21} denote the left hand side of Eq. (21). We prove Eq. (21) as follows.

$$\begin{aligned} \mathrm{LHS}_{21} &\leq \textstyle\sum_{h\in V_{\text{elim}}} p_H(h)^2/p_H(V(\Psi)) && [\text{since } p_H(V(\Psi[x/o])) \leq p_H(V(\Psi))] \\ &\leq \textstyle\sum_{h\in V_{\text{elim}}} p_H(h)\cdot p_H(V(\Psi))/p_H(V(\Psi)) && [\text{since } h\in V(\Psi) \Rightarrow p_H(h) \leq p_H(V(\Psi))] \\ &= p_H(V_{\text{elim}}) \end{aligned}$$

We conclude that f is adaptive submodular and strongly adaptive monotone. Additionally, instances with general priors are self–certifying for the same reason instances with uniform priors are, namely that if $\Phi_h \sim \Psi$ and $f(\mathrm{dom}(\Psi), \Phi_h) = f(X, \Phi_h)$ then $V(\Psi) = \{h\}$ and Φ_h is the only realization consistent with Ψ, which means that the instance satisfies the self–certifying condition. Hence we can apply Theorem 10 to this self–certifying instance with maximum reward threshold $Q = 1$, and minimum gap $\eta = 1/\min_h p_H(h)$, to obtain an upper bound of $\mathsf{OPT}\,(\ln(1/\min_h p_H(h)) + 1)$ on the number of queries made by the generalized binary search algorithm.

To improve this to an $\mathcal{O}(\log|H|)$-approximation in the event that $\min_h p_H(h)$ is extremely small using the observation of Kosaraju et al. (1999), call a policy π *progressive* if it eliminates at least one hypotheses from its version space in each query. Let $p'_H(h) = \max\left\{p_H(h), 1/|H|^2\right\}/\sum_{h'}\max\left\{p_H(h'), 1/|H|^2\right\}$ be the modified prior. Let $c(\pi, h)$ be the cost (i.e., # of queries) of π under target h. Then $c_{\text{avg}}(\pi, p) := \sum_h c(\pi, h)p(h)$ is the expected cost of π under prior p. We will show that $c_{\text{avg}}(\pi, p'_H)$ is a good approximation to $c_{\text{avg}}(\pi, p_H)$. Call h *rare* if $p_H(h) < 1/|H|^2$, and *common* otherwise. First, note that $\sum_{h'}\max\left\{p_H(h'), 1/|H|^2\right\} \leq 1 + 1/|H|$, and so $p'_H(h) \geq \frac{|H|}{|H|+1}p_H(h)$, for all h. Hence for all π, we have $c_{\text{avg}}(\pi, p'_H) \geq \frac{|H|}{|H|+1}c_{\text{avg}}(\pi, p_H)$. Next, we show $c_{\text{avg}}(\pi, p'_H) \leq c_{\text{avg}}(\pi, p_H) + 1$. Consider $c_{\text{avg}}(\pi, p'_H) - c_{\text{avg}}(\pi, p_H) = \sum_h c(\pi, h)\left(p'_H(h) - p_H(h)\right)$. The positive contributions must come from rare hypotheses. However, the total probability mass of these under p'_H is at most $1/|H|$, and since π is progressive we have $c(\pi, h) \leq |H|$ for all h, hence the difference in costs is at most one. Let $\alpha := \ln\left(\frac{1}{\min_h p'_H(h)}\right) + 1 \leq \ln\left(|H|^2 + |H|\right) + 1$ be the approximation factor for generalized binary search when run on p'_H. Let π be the policy of generalized binary search, and let π^* be an optimal policy under prior p_H. Then

$$c_{\text{avg}}(\pi, p_H) \leq \frac{|H|+1}{|H|}\, c_{\text{avg}}(\pi, p'_H) \leq \frac{|H|+1}{|H|}\,\alpha\, c_{\text{avg}}(\pi^*, p'_H) \leq \frac{|H|+1}{|H|}\,\alpha\, \left(c_{\text{avg}}(\pi^*, p_H) + 1\right)$$

With some further algebra, we can derive $c_{\text{avg}}(\pi, p_H) \leq (c_{\text{avg}}(\pi^*, p_H) + 1)\left(\ln\left(2e|H|^2\right)\right)$. Thus for a general prior a simple modification of generalized binary search yields an $\mathcal{O}(\log|H|)$-approximation. ■

Extensions to Arbitrary Costs, Multiple Classes, and Approximate Greedy Policies.

This result easily generalizes to handle the multi-class setting (i.e., $|O| \geq 2$), and α-approximate greedy policies, where we lose a factor of α in the approximation factor. As we describe in the Appendix, we can generalize adaptive submodularity to incorporate costs on items, which allows us to extend this result to handle query costs as well. We can therefore recover these extensions of Guillory and Bilmes (2009), while improving the approximation factor for GBS with item costs to $\ln\left(\frac{1}{\min_h p_H(h)}\right) + 1$. Guillory and Bilmes also showed how to extend the technique of Kosaraju et al. (1999) to obtain an $\mathcal{O}\left(\log\left(|H|\frac{\max_x c(x)}{\min_x c(x)}\right)\right)$-approximation with costs using a greedy policy, which may be combined with our tighter analysis as well to give a similar result with an improved leading constant. Recently, Gupta et al. (2010) showed how to simultaneously remove the dependence on both costs and probabilities from the approximation ratio. Specifically, within the context of studying an adaptive travelling salesman problem they investigated the *Optimal Decision Tree* problem, which is equivalent to the active learning problem we consider here. Using a clever, more complex algorithm than adaptive greedy, they achieve an $\mathcal{O}(\log|H|)$-approximation in the case of non-uniform costs and general priors.

10. Experiments

Greedy algorithms are often straightforward to develop and implement, which explains their popular use in practical applications, such as Bayesian experimental design and Active Learning, as discussed in §9 (also see the excellent introduction of Nowak (2009)) and Adaptive Stochastic Set Cover, e.g., for filter design in streaming databases as discussed in §7. Besides allowing us to prove approximation guarantees for such algorithms, adaptive submodularity provides the following immediate practical benefits:

1. The ability to use lazy evaluations to speed up its execution.
2. The ability to generate data-dependent bounds on the optimal value.

In this section, we empirically evaluate their benefits within a sensor selection application, in a setting similar to the one described by Deshpande et al. (2004). In this application, we have deployed a network V of wireless sensors, e.g., to monitor temperature in a building or traffic in a road network. Since sensors are battery constrained, we must adaptively select k sensors, and then, given those sensor readings, predict, e.g., the temperature at all remaining locations. This prediction is possible since temperature measurements will typically be correlated across space. Here, we will consider the case where sensors can fail to report measurements due to hardware failures, environmental conditions or interference.

The Sensor Selection Problem with Unreliable Sensors. More formally, we imagine every location $v \in V$ is associated with a random variable $\mathcal{X}_v$ describing the temperature at that location, and there is a joint probability distribution $\mathbb{P}[\mathcal{X}_V]$ that models the correlation between temperature values. Here, $\mathcal{X}_V = [\mathcal{X}_1, \dots, \mathcal{X}_n]$ is the random vector over all temperature values. We follow Deshpande et al. (2004) and assume that the joint distribution of the sensors is multivariate Gaussian. A sensor v can make a noisy observation $\mathcal{Y}_v = \mathcal{X}_v + \varepsilon_v$, where ε_v is zero mean Gaussian noise with known variance σ^2. If some measurements $\mathcal{Y}_A = \mathbf{y}_A$ are obtained at a subset of locations, then the conditional distribution $\mathbb{P}[\mathcal{X}_V \mid \mathcal{Y}_A = \mathbf{y}_A]$ allows predictions at the unobserved locations, e.g., by predicting $\mathbb{E}[\mathcal{X}_V \mid \mathcal{Y}_A = \mathbf{y}_A]$ (which minimizes the mean squared error). Furthermore, this conditional distribution quantifies the *uncertainty* in the prediction: Intuitively, we would like to select sensors that minimize the predictive uncertainty. One way to quantify the predictive uncertainty is to use the remaining Shannon entropy

$$\mathbb{H}(\mathcal{X}_V \mid \mathcal{Y}_A = \mathbf{y}_A) := \mathbb{E}\left[-\log_2\left(\mathbb{P}[\mathcal{X}_V \mid \mathcal{Y}_A = \mathbf{y}_A]\right)\right].$$

We would like to adaptively select k sensors, to maximize the expected reduction in Shannon entropy (e.g., Sebastiani and Wynn (2000)). However, in practice, sensors are often unreliable, and might fail to report their measurements. We assume that after selecting a sensor, we find out whether it has failed or not before deciding which sensor to select next. We suppose that each sensor has an associated probability $p_{\text{fail}}(v)$ of failure, in which case no reading is reported, and that sensor failures are independent of each other and of the ambient temperature at v. Thus we have an instance of the Stochastic Maximization problem with $E := V$, $O := \{\text{working}, \text{failed}\}$, and

$$f(A, \Phi) := \mathbb{H}(\mathcal{X}_V) - \mathbb{H}\left(\mathcal{X}_V \mid \mathbf{y}_{\{v \,:\, \Phi(v)=\text{working}\}}\right). \tag{22}$$

For multivariate normal distributions, the entropy is given as

$$\mathbb{H}(\mathcal{X}_V \mid \mathcal{Y}_A = \mathbf{y}_A) = \frac{1}{2}\ln(2\pi e)^n \left|\Sigma_{VA}\left(\Sigma_{AA} + \sigma^2 I\right)^{-1}\Sigma_{AV}\right|,$$

where for sets A and B, Σ_{AB} denotes the covariance (matrix) between random vectors $\mathcal{X}_A$ and $\mathcal{X}_B$. Note that the predictive covariance does not depend on the actual observations $\mathbf{y}_{\mathcal{A}}$, only on the set A of chosen locations. Thus,

$$\mathbb{H}(\mathcal{X}_V \mid \mathcal{Y}_A = \mathbf{y}_A) = \mathbb{H}(\mathcal{X}_V \mid \mathcal{Y}_A),$$

where as usual, $\mathbb{H}(\mathcal{X}_V \mid \mathcal{Y}_A) = \mathbb{E}\left[\mathbb{H}(\mathcal{X}_V \mid \mathcal{Y}_A = \mathbf{y}_A)\right]$. As Krause and Guestrin (2005) show, the function

$$g(A) := \mathbb{I}(\mathcal{X}_V; \mathcal{Y}_A) = \mathbb{H}(\mathcal{X}_V) - \mathbb{H}(\mathcal{X}_V \mid \mathcal{Y}_A) \tag{23}$$

is monotone submodular, whenever the observations $\mathcal{Y}_V$ are conditionally independent given $\mathcal{X}_V$.

This insight allows us to apply the result of §6 to show that the objective f defined in Eq. (22) is adaptive monotone submodular, using $\hat{f}(S) := g(\{v : (v, \text{working}) \in S\})$ for any $S \subseteq E \times O$.

Data and Experimental Setup. Our first data set consists of temperature measurements from the network of 46 sensors deployed at Intel Research Berkeley, which were sampled at 30 second intervals for 5 consecutive days (starting Feb. 28^{th}, 2004). We define our objective function with respect to the empirical covariance estimated from the data.

We also use data from traffic sensors deployed along the highway I-880 South in California. We use traffic speed data for all working days from 6 AM to 11 AM for one month, from 357 sensors. The goal is to predict the speed on all 357 road segments. We again estimate the empirical covariance matrix.

The Benefits of Lazy Evaluation. For both data sets, we run the adaptive greedy algorithm, using both the naive implementation (Algorithm 1) and the accelerated version using lazy evaluations (Algorithm 2). We vary the probability of sensor failure, and compute the number of evaluations of the function g (defined in Eq. (23)) each algorithm makes. (These function evaluations are the bottleneck in the computation, so the number of them serves as a machine-independent proxy for the running time.) Figures 5(a) and 5(c) show the performance. On the temperature data set, lazy evaluations speed up the computation by a factor of between roughly 3.5 and 7, depending on the failure probability. On the larger traffic data set, we obtain speedup factors between 30 and 38. We find that the benefit of the lazy evaluations increases with the problem size and with the failure probability. The dependence on problem size must ultimately be explained in terms of structural properties of the instances, which also benefit the nonadaptive accelerated greedy algorithm. The dependence on failure probability has a simpler explanation. Note that in these applications, if the accelerated greedy algorithm selects v, which then fails, then it does not need to make any additional function evaluations to select the next sensor. Contrast this with the naive greedy algorithm, which makes a function evaluation for each sensor that has not been selected so far.

The Benefits of the Data Dependent Bound. While adaptive submodularity allows us to prove worst-case performance guarantees for the adaptive greedy algorithm, in many practical applications it can be expected that these bounds are quite loose. For our sensor selection application, we use the data dependent bounds of Lemma 6 to compute an upper bound β_{avg} on $\max_\pi f_{\text{avg}}(\pi_{[k]})$ as described below, and compare it with the performance guarantee of Theorem 5. For the accelerated greedy algorithm, we use the upper bounds on the marginal benefits stored in the priority queue instead of recomputing the marginal benefits, and thus expect somewhat looser bounds. We find that for our application, the bounds are tighter than the worst case bounds. We also find that the "lazy" data dependent bounds are almost as tight as the "eager" bounds using the eagerly recomputed marginal benefits $\Delta(e \mid \Psi)$ for the latest and greatest Ψ. Figures 5(b) and 5(d) show the performance of the greedy algorithm as well as the three bounds on the optimal value.

Two subtleties arise when using the data-dependent bounds to bound $\max_\pi f_{\text{avg}}(\pi_{[k]})$. The first is that Lemma 6 tells us that $\Delta\left(\pi^*_{[k]} \mid \Psi\right) \leq \max_{A \subseteq E, |A| \leq k} \sum_{e \in A} \Delta(e \mid \Psi)$, whereas we would like to bound the difference between the optimal reward and the algorithm's current expected reward, conditioned on seeing Ψ, i.e., $\mathbb{E}\left[f(E(\pi^*_{[k]}, \Phi)) - f(\text{dom}(\Psi), \Phi) \;\mid\; \Phi \sim \Psi\right]$. However, in our applications f is strongly adaptive monotone, and strong adaptive monotonicity implies that for any π^* we have

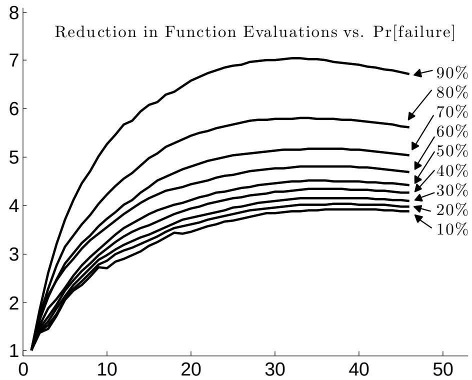
$$\mathbb{E}\left[f(E(\pi^*_{[k]}, \Phi)) - f(\text{dom}(\Psi), \Phi) \;\mid\; \Phi \sim \Psi\right] \leq \Delta\left(\pi^*_{[k]} \mid \Psi\right). \tag{24}$$

Hence, if we let $\mathsf{OPT}(\Psi) := \max_\pi \mathbb{E}\left[f(E(\pi_{[k]}, \Phi)) \;\mid\; \Phi \sim \Psi\right]$, Lemma 6 implies that

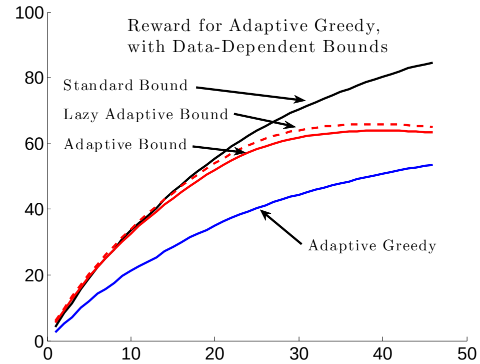
$$\mathsf{OPT}(\Psi) \leq \mathbb{E}\left[f(\text{dom}(\Psi), \Phi) \;\mid\; \Phi \sim \Psi\right] + \max_{A \subseteq E, |A| \leq k} \sum_{e \in A} \Delta(e \mid \Psi). \tag{25}$$

The second subtlety is that we obtain a sequence of bounds from Eq. (25). If we consider the (random) sequence of partial realizations observed by the adaptive greedy algorithm, $\emptyset = \Psi_0 \subset \Psi_1 \subset \cdots \subset \Psi_k$, we obtain $k+1$ bounds $\beta_0, \ldots, \beta_k$, where $\beta_i := \mathbb{E}\left[f(\text{dom}(\Psi_i), \Phi) \;\mid\; \Phi \sim \Psi_i\right] + \max_{A \subseteq E, |A| \leq k} \sum_{e \in A} \Delta(e \mid \Psi_i)$. Taking the expectation over Φ, note that for any π, and any i,

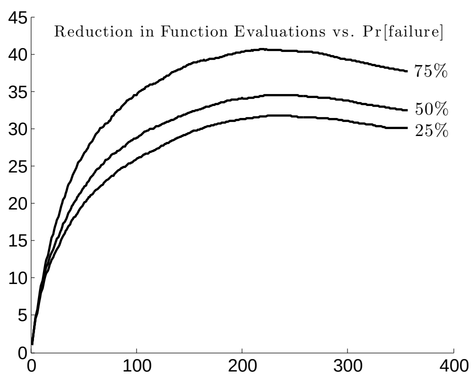
$$f_{\text{avg}}(\pi_{[k]}) \leq \mathbb{E}\left[\mathsf{OPT}(\Psi_i)\right] \leq \mathbb{E}\left[\beta_i\right].$$



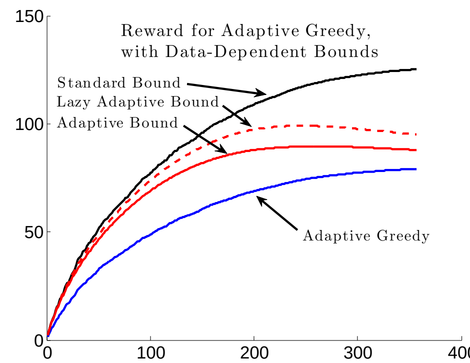
(a) *Temperature Data: The ratio of function evaluations made by the naive vs accelerated implementations of adaptive greedy vs. the budget k on number of sensors selected, for various failure rates. Averaged over* 100 *runs.*



(b) *Temperature Data: Rewards & bounds on the optimal value when $p_{fail}(v) = 0.5$ for all v vs. the budget k on number of sensors selected. Averaged over* 100 *runs.*



(c) *Traffic Data: The ratio of function evaluations made by the naive vs accelerated implementations of adaptive greedy vs. the budget k on number of sensors selected, for various failure rates. Averaged over* 10 *runs.*



(d) *Traffic Data: Rewards & bounds on the optimal value when $p_{fail}(v) = 0.5$ for all v vs. the budget k on number of sensors selected. Averaged over* 10 *runs.*

Figure 5: Experimental results.

Therefore for any $0 \leq i \leq k$, β_i is a random variable whose *expectation* is an upper bound on the optimal expected reward of any policy. At this point we may be tempted to use the minimum of these, i.e., $\beta_{\min} := \min_i \{\beta_i\}$ as our ultimate bound. However, a collection of random variables $X_0, \ldots, X_k$ with $\mathbb{E}[X_i] \geq \tau$ for all i does not, in general, satisfy $\min_i \{X_i\} \geq \tau$. While it is possible in our case, with its independent sensor failures, to use concentration inequalities to bound $\min_i \{\beta_i\} - \min_i \{\mathbb{E}[\beta_i]\}$ with high probability, and thus add an appropriate term to obtain a true upper bound from $\beta_{\min}$, we take a different approach; we simply use the average bound $\beta_{\text{avg}} := \frac{1}{k+1} \sum_{i=0}^{k} \beta_i$. Of course, depending on the application, a particular bound β_i (chosen independently of the sequence $\Psi_0, \Psi_1, \ldots, \Psi_k$) may be superior. For example, if g is modular, then β_0 is best, whereas if g exhibits strong diminishing returns, then bounds β_i with larger values of i may be significantly tighter.

11. Adaptivity Gap

An important question in adaptive optimization is *how much better* adaptive policies can perform when compared to non-adaptive policies. This is quantified by the *adaptivity gap*, which is the worst-case ratio, over problem instances, of the performance of the optimal adaptive policy to the optimal non-adaptive solution. Asadpour et al. (2008) show that in the Stochastic Submodular Maximization problem with independent failures (as considered in §6), the expected value of the optimal non-adaptive policy is at most a constant factor $1 - 1/e$ worse than the expected value of the optimal adaptive policy. While we currently do not have lower bounds for the adaptivity gap of the general Adaptive Stochastic Maximization problem (1), we can show that even in the case of adaptive submodular functions, the min-cost cover and min-sum cover versions have large adaptivity gaps, and thus there is a large benefit of using adaptive algorithms. In these cases, the adaptivity gap is defined as the worst-case ratio of the expected cost of the optimal non-adaptive policy divided by the expected cost of the optimal adaptive policy. For the Adaptive Stochastic Minimum Cost Coverage problem (2), Goemans and Vondrák (2006) show the special case of Stochastic Set Coverage without multiplicities has an adaptivity gap of $\Omega(|E|)$. Below we exhibit an adaptive stochastic optimization instance with adaptivity gap of $\Omega(|E|/\log|E|)$ for the Adaptive Stochastic Min-Sum Cover problem (3), which also happens to have the same adaptivity gap for Adaptive Stochastic Minimum Cost Coverage.

Theorem 19 *Even for adaptive submodular functions, the adaptivity gap of Adaptive Stochastic Min-Sum Cover is* $\Omega(n/\log n)$*, where* $n = |E|$.

Proof Suppose $E = \{1, \ldots, n\}$. Consider the active learning problem where our hypotheses $h : E \to \{-1, 1\}$ are threshold functions, i.e., $h(e) = 1$ if $e \geq \ell$ and $h(e) = -1$ if $e < \ell$ for some threshold ℓ. There is a uniform distribution over thresholds $\ell \in \{1, \ldots, n+1\}$. In order to identify the correct hypothesis with threshold ℓ, our policy must observe at least one of $\ell - 1$ or ℓ (typically both for $1 < \ell \leq n$). Let π_N be any non-adaptive policy, which can be represented as a permutation of E. It can be seen that the optimal non-adaptive policy must be a permutation; observing the same element multiple times can only increase the cost, and each element must eventually be selected to guarantee coverage. For the min-sum cover objective, consider playing π_N for $n/2$ time steps. It can be seen that the probability that the correct hypothesis has been identified is less than $1/2$. Thus a lower bound on the expected cost of π_N is $n/4$, since for $n/2$ time steps, at each time step a cost of at least $1/2$ is incurred. Thus, for both the min-cost and min-sum cover objectives the cost of the optimal non-adaptive policy is $\Omega(n)$.

As an example adaptive policy, we can implement a natural binary search strategy, which is guaranteed to identify the correct hypothesis after $O(\log n)$ steps, thus incurring cost $O(\log n)$, proving an adaptivity gap of $\Omega(n/\log n)$. ■

12. Hardness of Approximation

In this paper, we have developed the notion of adaptive submodularity, which characterizes when certain adaptive stochastic optimization problems are well-behaved in the sense that a simple greedy policy obtains a constant factor or logarithmic factor approximation to the best policy.

In contrast, we can also show that without adaptive submodularity, the adaptive stochastic optimization problems (1), (2), and (3) are extremely inapproximable, even with (pointwise) *modular* objective functions (i.e., those where for each Φ, $f : 2^E \times O^E \to \mathbb{R}$ is modular/linear in the first argument): We cannot hope to achieve an $\mathcal{O}(|E|^{1-\varepsilon})$ approximation ratio for these problems, unless the polynomial hierarchy collapses down to Σ_2^P.

Theorem 20 *In general, for all (possibly non-constant)* $\beta \geq 1$*, no polynomial time algorithm for Adaptive Stochastic Maximization with a budget of* βk *items can approximate the reward of an optimal policy with a budget of only* k *items to within a multiplicative factor of* $\mathcal{O}(|E|^{1-\varepsilon}/\beta)$ *for any* $\varepsilon > 0$*, unless PH =* Σ_2^P*. This holds even for pointwise linear* f.

We provide the proof of Theorem 20 in Appendix A.6. Note that by setting $\beta = 1$, we obtain $\mathcal{O}(|E|^{1-\varepsilon})$ hardness for Adaptive Stochastic Maximization. It turns out that in the instance distribution we construct in the proof of Theorem 20 the optimal policy covers every realization (i.e., always finds the treasure) using a budget of $k = \mathcal{O}(|E|^{\varepsilon/2})$ items. Hence if PH $\neq \Sigma_2^P$ then any randomized polynomial time algorithm wishing to cover this instance must have a budget $\beta = \Omega(|E|^{1-\varepsilon})$ times larger than the optimal policy, in order to ensure the ratio of rewards, which is $\Omega(|E|^{1-\varepsilon}/\beta)$, equals one. This yields the following corollary.

Corollary 21 *In general, no polynomial time algorithm for Adaptive Stochastic Min Cost Coverage can approximate the cost of an optimal policy to within a multiplicative factor of $\mathcal{O}(|E|^{1-\varepsilon})$ for any $\varepsilon > 0$, unless PH $= \Sigma_2^P$. This holds even for pointwise linear f.*

Furthermore, since in the instance distribution we construct the optimal policy π^* covers every realization using a budget of k, it has $c_\Sigma(\pi^*) \leq k$. Moreover, since we have shown that under our complexity theoretic assumptions, any polynomial time randomized policy π with budget βk achieves at most $o(\beta/|E|^{1-\varepsilon})$ of the (unit) value obtained by the optimal policy with budget k, it follows that $c_\Sigma(\pi) = \Omega(\beta k)$. Since we require $\beta = \Omega(|E|^{1-\varepsilon})$ to cover any set of realizations constituting, e.g., half of the probability mass, we obtain the following corollary.

Corollary 22 *In general, no polynomial time algorithm for Adaptive Stochastic Min-Sum Cover can approximate the cost of an optimal policy to within a multiplicative factor of $\mathcal{O}(|E|^{1-\varepsilon})$ for any $\varepsilon > 0$, unless PH $= \Sigma_2^P$. This holds even for pointwise linear f.*

13. Related Work

There is a large literature on adaptive optimization under partial observability which relates to adaptive submodularity, which can be broadly organized into several different categories. Here, we only review relevant related work that is not already discussed elsewhere in the manuscript.

Adaptive Versions of Classic Non-adaptive Optimization Problems. Many approaches consider stochastic generalizations of specific classic non-adaptive optimization problems, such as Set Cover (Goemans and Vondrák, 2006; Liu et al., 2008), Knapsack (Dean et al., 2008, 2005) and Traveling Salesman (Gupta et al., 2010). In contrast, in this paper our goal is to introduce a general problem structure – adaptive submodularity – that unifies a number of adaptive optimization problems. This is similar to how the classic notion of submodularity unifies various optimization problems such as Set Cover, Facility Location, nonadaptive Bayesian Experimental Design, etc.

Interactive Submodular Set Cover. Recent work by Guillory and Bilmes (2010) considers a class of adaptive optimization problems over a family of submodular objectives $\{f_h : h \in H\}$. In their problem, one must cover a monotone submodular objective f_{h^*} which depends on the (initially unknown) target hypothesis $h^* \in H$, by adaptively issuing queries and getting responses. Unlike traditional pool-based active learning, each query may generate a response from a *set* of valid responses depending on the target hypothesis. The reward is calculated by evaluating f_{h^*} on the set of (query, response) pairs observed, and the goal is to obtain some threshold Q of objective value at minimum total query cost, where queries may have nonuniform costs. Guillory and Bilmes consider the worst-case policy cost, and provide a greedy algorithm optimizing a clever hybrid objective function and prove it has an approximation guarantee of $\ln(Q|H|) + 1$ for integer valued objective functions $\{f_h\}_{h \in H}$.

While similar in spirit to this work, there are several significant differences between the two. Guillory and Bilmes focus on worst-case policy cost, while we focus mainly on average-case policy cost. The structure of adaptive submodularity depends on $\mathbb{P}[\Phi]$, whereas there is no such dependence in Interactive Submodular Set Cover. This dependence in turn allows us to obtain results, such as Theorem 10 for self–certifying instances, whose approximation guarantee does not depend on the number of realizations in the way that the guarantees for Interactive Submodular Set Cover depend on $|H|$. As Guillory and Bilmes prove, the latter dependence is

fundamental under reasonable complexity-theoretic assumptions[3]. An interesting open problem within the adaptive submodularity framework that is highlighted by the work on Interactive Submodular Set Cover is to identify useful instance-specific properties that are sufficient to improve upon the worst-case approximation guarantee of Theorem 11.

Greedy Frameworks for Adaptive Optimization. The paper that is perhaps closest in spirit to this work is the one on Stochastic Depletion problems by Chan and Farias (2009), who also identify a general class of adaptive optimization problems than can be near-optimally solved using greedy algorithms (which in their setting gives a factor 2 approximation). However, the similarity is mainly on a conceptual level: The problems and approaches, as well as example applications considered, are quite different.

Stochastic Optimization with Recourse. A class of adaptive optimization problems studied extensively in operations research (since Dantzig (1955)) is the area of *stochastic optimization with recourse*. Here, an optimization problem, such as Set Cover, Steiner Tree or Facility Location, is presented in multiple stages. At each stage, more information is revealed, but costs of actions increase. A key difference to the problems studied in this paper is that in these problems, information gets revealed independently of the actions taken by the algorithm. There are general efficient, sampling based (approximate) reductions of multi-stage optimization to the deterministic setting, see, e.g., Gupta et al. (2005).

Bayesian Global Optimization. Adaptive Stochastic Optimization is also related to the problem of Bayesian Global Optimization (*c.f.*, Brochu et al. (2009) for a recent survey of the area). In Bayesian Global Optimization, the goal is to adaptively select inputs in order to maximize an unknown function that is expensive to evaluate (and can possibly only be evaluated using noisy observations). A common approach that has been successful in many applications (*c.f.*, Lizotte et al. (2007) for a recent application in machine learning), is to assume a prior distribution, such as a Gaussian process, over the unknown objective function. Several criteria for selecting inputs have been developed, such as the Expected Improvement (Jones et al., 1998) criterion. However, while recently performance guarantees where obtained in the no-regret setting (Grünewälder et al., 2010; Srinivas et al., 2010), we are not aware of any approximation guarantees for Bayesian Global Optimization.

Probabilistic Planning. The problem of decision making under partial observability has also been extensively studied in stochastic optimal control. In particular, Partially Observable Markov Decision Processes (POMDPs, Smallwood and Sondik (1973)) are a general framework that capture many adaptive optimization problems under partial observability. Unfortunately, solving POMDPs is PSPACE hard (Papadimitriou and Tsitsiklis, 1987), thus typically heuristic approximations with no performance guarantees are applied (Pineau et al., 2006). For some special instances of POMDPs related to Multi-armed Bandit problems, (near-)optimal policies can be found. These include the (optimal) Gittins-index policy for the classic Multi-armed Bandit problem (Gittins and Jones, 1979) and approximate policies for the Multi-armed Bandit problem with metric switching costs (Guha and Munagala, 2009) and special cases of the Restless Bandit problem (Guha et al., 2009). The problems considered in this paper can be formalized as POMDPs, albeit with exponentially large state space (where the state represents the selected items and observations). Thus our results can be interpreted as widening the class of partially observable optimization problems that can be efficiently approximately solved.

Previous Work by the Authors. This manuscript is an extended version of a paper that appeared in the Conference on Learning Theory (COLT) 2010 (Golovin and Krause, 2010). The present version significantly expands on the previous one, and includes an improved policy-centric treatment of adaptive submodularity, new theoretical results on adaptive coverage and min-sum coverage, the accelerated adaptive greedy algorithm, new applications, a new kitchen sink, new hardness and adaptivity gap results, the incorporation of item costs, and experiments.

3. They reduce to Set Cover and use the result of Feige (1998), which requires the assumption NP $\not\subseteq$ DTIME($n^{\mathcal{O}(\log\log n)}$), but it suffices to assume only P $\neq$ NP using the Set Cover approximation hardness result of Raz and Safra (1997) instead.

14. Conclusions

In this paper, we introduced the concept of *adaptive submodularity*, generalizing submodular set functions to adaptive policies. Our generalization is based on a natural adaptive analog of the diminishing returns property well understood for set functions. In the special case of deterministic distributions, adaptive submodularity reduces to the classical notion of submodular set functions. We proved that several guarantees carried by the non-adaptive greedy algorithm for submodular set functions generalize to a natural adaptive greedy algorithm in the case of adaptive submodular functions, for constrained maximization and certain natural coverage problems with both minimum cost and minimum sum objectives. We also showed how the adaptive greedy algorithm can be accelerated using lazy evaluations, and how one can compute data-dependent bounds on the optimal solution. We illustrated the usefulness of the concept by giving several examples of adaptive submodular objectives arising in diverse applications including sensor placement, viral marketing and pool-based active learning. Proving adaptive submodularity for these problems allowed us to recover existing results in these applications as special cases and lead to natural generalizations. Our experiments on real data indicate that adaptive submodularity can provide practical benefits, such as significant speed ups and tighter data-dependent bounds. We believe that our results provide an interesting step in the direction of exploiting structure to solve complex stochastic optimization problems under partial observability.

Acknowledgments

This research was partially supported by ONR grant N00014-09-1-1044, NSF grant CNS-0932392, NSF grant IIS-0953413, a gift by Microsoft Corporation, an Okawa Foundation Research Grant, and by the Caltech Center for the Mathematics of Information.

References

Esther M. Arkin, Henk Meijer, Joseph S. B. Mitchell, David Rappaport, and Steven S. Skiena. Decision trees for geometric models. In *Proc. of Symposium on Computational Geometry*, pages 369–378, New York, NY, USA, 1993. ACM. ISBN 0-89791-582-8. doi: http://doi.acm.org/10.1145/160985.161167.

Arash Asadpour, Hamid Nazerzadeh, and Amin Saberi. Stochastic submodular maximization. In *WINE '08: Proc. of the 4th International Workshop on Internet and Network Economics*, pages 477–489, Berlin, Heidelberg, 2008. Springer-Verlag. ISBN 978-3-540-92184-4. doi: http://dx.doi.org/10.1007/978-3-540-92185-1_53.

Eric Brochu, Mike Cora, and Nando de Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. Technical Report TR-2009-23, Department of Computer Science, University of British Columbia, November 2009.

V. T. Chakaravarthy, V. Pandit, S. Roy, P. Awasthi, and M. Mohania. Decision trees for entity identification: Approximation algorithms and hardness results. In *In Proceedings of the ACM-SIGMOD Symposium on Principles of Database Systems*, 2007.

Carri W. Chan and Vivek F. Farias. Stochastic depletion problems: Effective myopic policies for a class of dynamic optimization problems. *Math. Oper. Res.*, 34(2):333–350, 2009. ISSN 0364-765X. doi: http://dx.doi.org/10.1287/moor.1080.0364.

David A. Cohn, Zoubin Gharamani, and Michael I. Jordan. Active learning with statistical models. *J AI Research*, 4:129–145, 1996.

George B. Dantzig. Linear programming under uncertainty. *Management Sci*, 1:197–206, 1955.

Sanjoy Dasgupta. Analysis of a greedy active learning strategy. In *NIPS: Advances in Neural Information Processing Systems*, pages 337–344. MIT Press, 2004.

B. Dean, M.X. Goemans, and J. Vondrák. Adaptivity and approximation for stochastic packing problems. In *Proceedings of the 16th ACM-SIAM Symposium on Discrete Algorithms,*, pages 395–404, 2005.

B. Dean, M.X. Goemans, and J. Vondrák. Approximating the stochastic knapsack problem: The benefit of adaptivity. *Mathematics of Operations Research*, 33:945–964, 2008.

A. Deshpande, C. Guestrin, S. Madden, J. Hellerstein, and W. Hong. Model-driven data acquisition in sensor networks. In *VLDB*, pages 588–599, 2004.

Uriel Feige. A threshold of ln n for approximating set cover. *Journal of the ACM*, 45(4):634 – 652, July 1998.

Uriel Feige and Carsten Lund. On the hardness of computing the permanent of random matrices. *Computational Complexity*, 6(2):101–132, 1997. ISSN 1016-3328. doi: http://dx.doi.org/10.1007/BF01262928.

Uriel Feige, László Lovász, and Prasad Tetali. Approximating min sum set cover. *Algorithmica*, 40(4): 219–234, 2004.

Satoru Fujishige. *Submodular functions and optimization*. Number 47. Annals of Discrete. Mathematics, North Holland, Amsterdam, 2nd edition, 1991. ISBN 0-444-88556-0.

M. R. Garey and Ronald L. Graham. Performance bounds on the splitting algorithm for binary testing. *Acta Inf.*, 3:347–355, 1974.

J. C. Gittins and D. M. Jones. A dynamic allocation index for the discounted multiarmed bandit problem. *Biometrika*, 66(3):561–565, December 1979.

Michel X. Goemans and Jan Vondrák. Stochastic covering and adaptivity. In *LATIN*, pages 532–543, 2006.

Daniel Golovin and Andreas Krause. Adaptive submodularity: A new approach to active learning and stochastic optimization. In *23rd Annual Conference on Learning Theory*, pages 333–345, 2010.

Daniel Golovin, Anupam Gupta, Amit Kumar, and Kanat Tangwongsan. All-norms and all-L_p-norms approximation algorithms. In Ramesh Hariharan, Madhavan Mukund, and V Vinay, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2008)*, Dagstuhl, Germany, 2008. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

Pranava R. Goundan and Andreas S. Schulz. Revisiting the greedy approach to submodular set function maximization. Technical report, Massachusetts Institute of Technology, 2007.

Steffen Grünewälder, Jean-Yves Audibert, Manfred Opper, and John Shawe-Taylor. Regret bounds for gaussian process bandit problems. In *Proc. of the 13th International Conference on Artificial Intelligence and Statistics*, 2010.

Sudipto Guha and Kamesh Munagala. Multi-armed bandits with metric switching costs. In *ICALP*, 2009.

Sudipto Guha, Kamesh Munagala, and Peng Shi. Approximation algorithms for restless bandit problems. Technical Report 0711.3861v5, arXiv, 2009.

Andrew Guillory and Jeff Bilmes. Average-case active learning with costs. In *The 20th International Conference on Algorithmic Learning Theory*, University of Porto, Portugal, October 2009.

Andrew Guillory and Jeff Bilmes. Interactive submodular set cover. In *Proc. International Conference on Machine Learning (ICML)*, number UWEETR-2010-0001, Haifa, Israel, 2010.

Anupam Gupta, Martin Pál, R. Ravi, and Amitabh Sinha. What about wednesday? approximation algorithms for multistage stochastic optimization. In *Proc. APPROX*, 2005.

Anupam Gupta, Ravishankar Krishnaswamy, Viswanath Nagarajan, and R. Ravi. Approximation algorithms for optimal decision trees and adaptive TSP problems. In *Proc. International Colloquium on Automata, Languages and Programming (ICALP)*, volume 6198 of *Lecture Notes in Computer Science*, pages 690–701. Springer, 2010. ISBN 978-3-642-14164-5.

Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455–492, 1998.

Haim Kaplan, Eyal Kushilevitz, and Yishay Mansour. Learning with attribute costs. In *Proceedings of the* 37^{th} *STOC*, pages 356–365, 2005.

David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, New York, NY, USA, 2003. ACM. ISBN 1-58113-737-0. doi: http://doi.acm.org/10.1145/956750.956769.

S. Rao Kosaraju, Teresa M. Przytycka, and Ryan S. Borgstrom. On an optimal split tree problem. In *Proc. of the 6th Intl. Workshop on Algorithms and Data Structures*, pages 157–168, London, UK, 1999. Springer-Verlag. ISBN 3-540-66279-0.

A. Krause and C. Guestrin. Near-optimal nonmyopic value of information in graphical models. In *Proceedings of Uncertainty in Artificial Intelligence (UAI)*, 2005.

Andreas Krause and Carlos Guestrin. Near-optimal observation selection using submodular functions. In *AAAI Nectar track*, pages 1650–1654, 2007.

Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance. Cost-effective outbreak detection in networks. In *KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 420–429, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-609-7. doi: http://doi.acm.org/10.1145/1281192.1281239.

M. Littman, J. Goldsmith, and M. Mundhenk. The computational complexity of probabilistic planning. *Journal of Artificial Intelligence Research*, 9:1–36, 1998.

Zhen Liu, Srinivasan Parthasarathy, Anand Ranganathan, and Hao Yang. Near-optimal algorithms for shared filter evaluation in data stream systems. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 133–146, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-102-6. doi: http://doi.acm.org/10.1145/1376616.1376633.

Daniel Lizotte, Tao Wang, Michael Bowling, and Dale Schuurmans. Automatic gait optimization with gaussian process regression. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 944–949, 2007.

Donald W. Loveland. Performance bounds for binary testing with arbitrary weights. *Acta Inf.*, 22(1):101–114, 1985. ISSN 0001-5903. doi: http://dx.doi.org/10.1007/BF00290148.

Andrew McCallum and Kamal Nigam. Employing EM and pool-based active learning for text classification. In *ICML*, pages 350–358, 1998.

Michel Minoux. Accelerated greedy algorithms for maximizing submodular set functions. In *Proc. of the 8th IFIP Conference on Optimization Techniques*, pages 234–243. Springer, 1978.

Kamesh Munagala, Shivnath Babu, Rajeev Motwani, Jennifer Widom, and Eiter Thomas. The pipelined set cover problem. In *Proc. Intl. Conf. on Database Theory*, pages 83–98, 2005.

George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions - I. *Mathematical Programming*, 14(1):265–294, 1978.

Rob Nowak. Noisy generalized binary search. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1366–1374. 2009.

C. H. Papadimitriou and J. N. Tsitsiklis. The complexity of markov decision processses. *Mathematics of Operations Research*, 12(3):441–450, 1987.

Joelle Pineau, Geoff Gordon, and Sebastian Thrun. Anytime point-based approximations for large POMDPs. *JAIR*, 27:335–380, 2006.

Ran Raz and Shmuel Safra. A sub-constant error–probability low–degree test, and a sub–constant error-probability PCP characterization of NP. In *STOC '97: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 475–484, New York, NY, USA, 1997. ACM. ISBN 0-89791-888-6. doi: http://doi.acm.org/10.1145/258533.258641.

Alexander Schrijver. *Combinatorial optimization : polyhedra and efficiency*. Volume B, Part IV, Chapters 39-49. Springer, 2003.

P. Sebastiani and H. P. Wynn. Maximum entropy sampling and optimal bayesian experimental design. *Journal of the Royal Statistical Society, Series B*, 62(1):145–157, 2000.

R.D. Smallwood and E.J. Sondik. The optimal control of partially observable Markov decision processes over a finite horizon. *Operations Research*, 21:1071–1088, 1973.

Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *In Proc. International Conference on Machine Learning (ICML)*, 2010.

Matthew Streeter and Daniel Golovin. An online algorithm for maximizing submodular functions. Technical Report CMU-CS-07-171, Carnegie Mellon University, 2007.

Matthew Streeter and Daniel Golovin. An online algorithm for maximizing submodular functions. In *NIPS: Advances in Neural Information Processing Systems*, pages 1577–1584, 2008.

Laurence A. Wolsey. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4):385–393, 1982.

Appendix A. Additional Proofs and Incorporating Item Costs

In this appendix we provide all of the proofs omitted from the main text. For the results of §5, we do so by first explaining how our results generalize to the case where items have costs, and then proving generalizations which incorporate item costs.

A.1 Incorporating Costs: Preliminaries

In this section we provide the preliminaries required to define and analyze the versions of our problems with non-uniform item costs. We suppose each item $e \in E$ has a cost $c(e)$, and the cost of a set $S \subseteq E$ is given by the modular function $c(S) = \sum_{e \in S} c(e)$. We define the generalizations of problems (1), (2), and (3) in §A.3, §A.4, and §A.5, respectively.

Our results are with respect to the greedy policy π^{greedy} and α-approximate greedy policies. With costs, the greedy policy selects an item maximizing $\Delta(e \mid \Psi)/c(e)$, where Ψ is the current partial realization.

Definition 23 (Approximate Greedy Policy with Costs) *A policy π is an α-*approximate greedy policy *if for all Ψ such that there exists $e \in E$ with $\Delta(e \mid \Psi) > 0$,*

$$\pi(\Psi) \in \left\{ e \,:\, \frac{\Delta(e \mid \Psi)}{c(e)} \;\geq\; \frac{1}{\alpha} \max_{e'} \left(\frac{\Delta(e' \mid \Psi)}{c(e')} \right) \right\},$$

and π terminates upon observing any Ψ such that $\Delta(e \mid \Psi) \leq 0$ for all $e \in E$. That is, an α-approximate greedy policy always obtains at least $(1/\alpha)$ of the maximum possible ratio of conditional expected marginal benefit to cost, and terminates when no more benefit can be obtained in expectation. A greedy policy *is any* 1*-approximate greedy policy.*

It will be convenient to imagine the policy executing over time, such that when a policy π selects an item e, it starts to *run* e, and *finishes running* e after $c(e)$ units of time. We next generalize the definition of policy truncation. Actually we require three such generalizations, which are all equivalent in the unit cost case.

Definition 24 (Strict Policy Truncation) *The* strict level t truncation *of a policy π, denoted by $\pi_{[\leftarrow t]}$, is obtained by running π for t time units, and unselecting items whose runs have not finished by time t. Formally, $\pi_{[\leftarrow t]}$ has domain $\left\{\Psi \in \mathrm{dom}(\pi) \ : \ c(\pi(\Psi)) + \sum_{e \in \mathrm{dom}(\Psi)} c(e) \leq t\right\}$, and agrees with π everywhere in its domain.*

Definition 25 (Lax Policy Truncation) *The* lax level t truncation *of a policy π, denoted by $\pi_{[t \rightarrow]}$, is obtained by running π for t time units, and selecting the items running at time t. Formally, $\pi_{[t \rightarrow]}$ has domain $\left\{\Psi \in \mathrm{dom}(\pi) \ : \ \sum_{e \in \mathrm{dom}(\Psi)} c(e) < t\right\}$, and agrees with π everywhere in its domain.*

Definition 26 (Policy Truncation with Costs) *The* level-t-truncation *of a policy π, denoted by $\pi_{[t]}$, is a randomized policy obtained by running π for t time units, and if some item e has been running for $0 \leq \tau < c(e)$ time at time t, selecting e independently with probability $\tau/c(e)$. Formally, $\pi_{[t]}$ is a randomized policy that agrees with π everywhere in its domain, has $\mathrm{dom}(\pi_{[\leftarrow t]}) \subseteq \mathrm{dom}(\pi_{[t]}) \subseteq \mathrm{dom}(\pi_{[t \rightarrow]})$ with certainty, and includes each $\Psi \in \mathrm{dom}(\pi_{[t \rightarrow]}) \setminus \mathrm{dom}(\pi_{[\leftarrow t]})$ in its domain independently with probability $\left(t - \sum_{e \in \mathrm{dom}(\Psi)} c(e)\right)/c(\pi(\Psi))$.*

In the proofs that follow, we will need a notion of the conditional expected cost of a policy, as well as an alternate characterization of adaptive monotonicity, based on a notion of policy concatenation. We prove the equivalence of our two adaptive monotonicity conditions in Lemma 30.

Definition 27 (Conditional Policy Cost) *The* conditional policy cost *of π conditioned on Ψ, denoted $c(\pi \mid \Psi)$, is the expected cost of the items π selects under $\mathbb{P}[\Phi \mid \Psi]$. That is, $c(\pi \mid \Psi) := \mathbb{E}[c(E(\pi, \Phi)) \mid \Phi \sim \Psi]$.*

Definition 28 (Policy Concatenation) *Given two policies π_1 and π_2 define $\pi_1 @ \pi_2$ as the policy obtained by running π_1 to completion, and then running policy π_2 as if from a fresh start, ignoring the information gathered*[4] *during the running of π_1.*

Definition 29 (Adaptive Monotonicity (Alternate Version)) *A function $f : 2^E \times O^E \rightarrow \mathbb{R}_{\geq 0}$ is* adaptive monotone *with respect to distribution $\mathbb{P}[\Phi]$ if for all policies π and π', it holds that $f_{avg}(\pi) \leq f_{avg}(\pi' @ \pi)$, where $f_{avg}(\pi) := \mathbb{E}[f(E(\pi, \Phi), \Phi)]$ is defined w.r.t. $\mathbb{P}[\Phi]$.*

Lemma 30 (Adaptive Monotonicity Equivalence) *Fix a function $f : 2^E \times O^E \rightarrow \mathbb{R}_{\geq 0}$. Then $\Delta(e \mid \Psi) \geq 0$ for all Ψ with $\mathbb{P}[\Psi] > 0$ and all $e \in E$ if and only if for all policies π and π', $f_{avg}(\pi) \leq f_{avg}(\pi' @ \pi)$.*

Proof Fix policies π and π'. We begin by proving $f_{\mathrm{avg}}(\pi' @ \pi) = f_{\mathrm{avg}}(\pi @ \pi')$. Fix any Φ and note that $E(\pi' @ \pi, \Phi) = E(\pi', \Phi) \cup E(\pi, \Phi) = E(\pi @ \pi', \Phi)$. Hence $f_{\mathrm{avg}}(\pi' @ \pi) = \mathbb{E}[f(E(\pi' @ \pi, \Phi), \Phi)] = \mathbb{E}[f(E(\pi @ \pi', \Phi), \Phi)] = f_{\mathrm{avg}}(\pi @ \pi')$. So the $f_{\mathrm{avg}}(\pi) \leq f_{\mathrm{avg}}(\pi' @ \pi)$ condition can be replaced with the condition $f_{\mathrm{avg}}(\pi) \leq f_{\mathrm{avg}}(\pi @ \pi')$.

4. Technically, if under any realization Φ policy π_2 selects an item that π_1 previously selected, then $\pi_1 @ \pi_2$ cannot be written as a function from a set of partial realizations to E, i.e., it is not a policy. This can be amended by allowing partial realizations to be multisets over elements of $E \times O$, so that, e.g., if e is played twice then $(e, \Phi(e))$ appears twice in Ψ. However, in the interest of readability we will avoid this more cumbersome multiset formalism, and abuse notation slightly by calling $\pi_1 @ \pi_2$ a policy. This issue arises whenever we run some policy and then run another from a fresh start.

We first prove the forward direction. Suppose $\Delta(e \mid \Psi) \geq 0$ for all Ψ and all $e \in E$. Note the expression $f_{\text{avg}}(\pi @ \pi') - f_{\text{avg}}(\pi)$ can be written as a conical combination of (nonnegative) $\Delta(e \mid \Psi)$ terms, i.e., for some $\alpha \geq \mathbf{0}$, $f_{\text{avg}}(\pi @ \pi') - f_{\text{avg}}(\pi) = \sum_{\Psi, e} \alpha_{(\Psi, e)} \Delta(e \mid \Psi)$. Hence $f_{\text{avg}}(\pi @ \pi') - f_{\text{avg}}(\pi) \geq 0$ and so $f_{\text{avg}}(\pi) \leq f_{\text{avg}}(\pi @ \pi') = f_{\text{avg}}(\pi' @ \pi)$.

We next prove the backward direction, in contrapositive form. Suppose $\Delta(e \mid \Psi) < 0$ for some Ψ with $\mathbb{P}[\Psi] > 0$ and $e \in E$. Let $e_1, \ldots, e_r$ be the items in $\mathrm{dom}(\Psi)$ and define policies π and π' as follows. For $i = 1, 2, \ldots, r$, both π and π' select e_i and observe $\Phi(e_i)$. If either policy observes $\Phi(e_i) \neq \Psi(e_i)$ it immediately terminates, otherwise it continues. If π succeeds in selecting all of $\mathrm{dom}(\Psi)$ then it terminates. If π' succeeds in selecting all of $\mathrm{dom}(\Psi)$ then it selects e and then terminates. We claim $f_{\text{avg}}(\pi @ \pi') - f_{\text{avg}}(\pi) < 0$. Note that $E(\pi @ \pi', \Phi) = E(\pi, \Phi)$ unless $\Phi \sim \Psi$, and if $\Phi \sim \Psi$ then $E(\pi @ \pi', \Phi) = E(\pi, \Phi) \cup \{e\}$ and also $E(\pi, \Phi) = \mathrm{dom}(\Psi)$. Hence

$$\begin{aligned}
f_{\text{avg}}(\pi @ \pi') - f_{\text{avg}}(\pi) &= \mathbb{E}\left[f(E(\pi @ \pi', \Phi), \Phi) - f(E(\pi, \Phi), \Phi)\right] \\
&= \mathbb{E}\left[f(E(\pi @ \pi', \Phi), \Phi) - f(E(\pi, \Phi), \Phi) \mid \Phi \sim \Psi\right] \cdot \mathbb{P}[\Phi \sim \Psi] \\
&= \mathbb{E}\left[f(\mathrm{dom}(\Psi) \cup \{e\}, \Phi) - f(\mathrm{dom}(\Psi), \Phi) \mid \Phi \sim \Psi\right] \cdot \mathbb{P}[\Phi \sim \Psi] \\
&= \Delta(e \mid \Psi) \cdot \mathbb{P}[\Phi \sim \Psi]
\end{aligned}$$

The last term is negative, as $\mathbb{P}[\Phi \sim \Psi] > 0$ and $\Delta(e \mid \Psi) < 0$ by assumption. Therefore $f_{\text{avg}}(\pi) > f_{\text{avg}}(\pi @ \pi') = f_{\text{avg}}(\pi' @ \pi)$, which completes the proof. ■

A.2 Adaptive Data Dependent Bounds with Costs

The adaptive data dependent bound has the following generalization with costs.

Lemma 31 (The Adaptive Data Dependent Bound with Costs) *Suppose we have made observations Ψ after selecting $\mathrm{dom}(\Psi)$. Let π^* be any policy. Then for adaptive monotone submodular $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$*

$$\Delta(\pi^* \mid \Psi) \;\leq\; Z \;\leq\; c(\pi^* \mid \Psi) \max_e \left(\frac{\Delta(e \mid \Psi)}{c(e)} \right) \tag{26}$$

where $Z = \max_w \left\{ \sum_{e \in E} w_e \Delta(e \mid \Psi) \;:\; \sum_e c(e) w_e \leq c(\pi^* \mid \Psi)\text{ and }\forall e \in E, 0 \leq w_e \leq 1 \right\}$.

Proof Order the items in $\mathrm{dom}(\Psi)$ arbitrarily, and consider the policy π that for each $e \in \mathrm{dom}(\Psi)$ in order selects e, terminating if $\Phi(e) \neq \Psi(e)$ and proceeding otherwise, and, should it succeed in selecting all of $\mathrm{dom}(\Psi)$ without terminating (which occurs iff $\Phi \sim \Psi$), then proceeds to run π^* as if from a fresh start, forgetting the observations in Ψ. By construction the expected marginal benefit of running the π^* portion of π conditioned on $\Phi \sim \Psi$ equals $\Delta(\pi^* \mid \Psi)$. For all $e \in E$, let $w(e) = \mathbb{P}[e \in E(\pi, \Phi) \mid \Phi \sim \Psi]$ be the probability that e is selected when running π, conditioned on $\Phi \sim \Psi$. Whenever some $e \in E \setminus \mathrm{dom}(\Psi)$ is selected by π, the current partial realization Ψ' contains Ψ as a subrealization; hence adaptive submodularity implies $\Delta(e \mid \Psi') \leq \Delta(e \mid \Psi)$. It follows that the total contribution of e to $\Delta(\pi^* \mid \Psi)$ is upper bounded by $w(e) \cdot \Delta(e \mid \Psi)$. Summing over $e \in E \setminus \mathrm{dom}(\Psi)$, we get a bound of $\Delta(\pi^* \mid \Psi) \leq \sum_{e \in E \setminus \mathrm{dom}(\Psi)} w(e) \Delta(e \mid \Psi)$. Next, note that each $e \in E \setminus \mathrm{dom}(\Psi)$ contributes $w(e)c(e)$ cost to $c(\pi^* \mid \Psi)$. Hence it must be the case that $\sum_{e \in E \setminus \mathrm{dom}(\Psi)} w(e)c(e) \leq c(\pi^* \mid \Psi)$. Obviously, $w(e) \in [0, 1]$ for all e, since $w(e)$ is a probability. Hence $\Delta(\pi^* \mid \Psi) \leq \sum_{e \in E \setminus \mathrm{dom}(\Psi)} w(e) \Delta(e \mid \Psi) \leq Z$ because setting $w_e = w(e)$ is feasible for the the linear program for which Z is the optimal value.

To show $Z \leq c(\pi^* \mid \Psi) \max_e (\Delta(e \mid \Psi) / c(e))$, consider any feasible solution w to the linear program defining Z. It attains objective value

$$\sum_{e \in E} w_e \Delta(e \mid \Psi) \leq \sum_{e \in E} w_e c(e) \frac{\Delta(e \mid \Psi)}{c(e)} \leq \sum_{e \in E} w_e c(e) \max_{e \in E} \left(\frac{\Delta(e \mid \Psi)}{c(e)} \right) \leq c(\pi^* \mid \Psi) \max_{e \in E} \left(\frac{\Delta(e \mid \Psi)}{c(e)} \right)$$

since $\sum_{e \in E} w_e c(e) \le c(\pi^* \mid \Psi)$ by the feasibility of w. ■

A simple greedy algorithm can be used to compute Z; we provide pseudocode for it in Algorithm 3. The correctness of this algorithm is more readily discerned upon rewriting the linear program using variables $x_e = c(e)w_e$ to obtain $Z = \max_x \left\{\sum_{e \in E} x_e \left(\Delta(e \mid \Psi)/c(e)\right) \;:\; \sum_e x_e \le c(\pi^* \mid \Psi) \text{ and } \forall e \in E, 0 \le x_e \le c(e)\right\}$. Intuitively, it is clear that to optimize x we should shift mass towards variables with the highest $\Delta(e \mid \Psi)/c(e)$ ratio. Clearly, any optimal solution has $\sum_e x_e = c(\pi^* \mid \Psi)$. Moreover, in any optimal solution, $\Delta(e \mid \Psi)/c(e) > \Delta(e' \mid \Psi)/c(e')$ implies $x_e = c(e)$ or $x_{e'} = 0$, since otherwise it would be possible to shift mass from $x_{e'}$ to x_e and obtain an increase in objective value. If the $\Delta(e \mid \Psi)/c(e)$ values are distinct for distinct items, there will be a unique solution satisfying these constraints, which Algorithm 3 will compute. Otherwise, we imagine perturbing each $\Delta(e \mid \Psi)$ by a independent random quantities ϵ_e drawn uniformly from $[0, \epsilon]$ to make them distinct. This changes the optimum value by at most $|E|\epsilon$, which vanishes as we let ϵ tend towards zero. Hence any solution satisfying $\sum_e x_e = c(\pi^* \mid \Psi)$ and $\Delta(e \mid \Psi)/c(e) > \Delta(e' \mid \Psi)/c(e')$ implies $x_e = c(e)$ or $x_{e'} = 0$ is optimal. Since Algorithm 3 outputs the value of such a solution, it is correct.

Input: Groundset E; Partial realization Ψ; Costs $c : E \to \mathbb{N}$; Budget $C = c(\pi^* \mid \Psi)$; Conditional expected marginal benefits $\Delta(e \mid \Psi)$ for all $e \in E$.
Output: $Z = \max_w \left\{\sum_{e \in E} w_e \, \Delta(e \mid \Psi) \;:\; \sum_e c(e) w_e \le c(\pi^* \mid \Psi) \text{ and } \forall e \in E, 0 \le w_e \le 1\right\}$
begin
  Sort E by $\Delta(e \mid \Psi)/c(e)$, so that $\frac{\Delta(e_1 \mid \Psi)}{c(e_1)} \ge \frac{\Delta(e_2 \mid \Psi)}{c(e_2)} \ge \ldots \ge \frac{\Delta(e_n \mid \Psi)}{c(e_n)}$;
  Set $w \leftarrow \mathbf{0}$; $i \leftarrow 0$; $a \leftarrow 0$; $z \leftarrow 0$; $e \leftarrow$ NULL;
  while $a < C$ **do**
    $i \leftarrow i + 1$; $e \leftarrow e_i$;
    $w_e \leftarrow \min\{1, C - a\}$;
    $a \leftarrow a + c(e) w_e$; $z \leftarrow z + w_e \Delta(e \mid \Psi)$;
  Output z;
end

Algorithm 3: Algorithm to compute the data dependent bound Z of Lemma 31.

A.3 The Max-Cover Objective

With item costs, the Adaptive Stochastic Maximization problem becomes one of finding some

$$\pi^* \in \arg\max_{\pi} f_{\text{avg}}(\pi_{[k]}) \tag{27}$$

where k is a budget on the cost of selected items, and we define $f_{\text{avg}}(\pi)$ for a randomized policy π to be $f_{\text{avg}}(\pi) := \mathbb{E}\left[f(E(\pi, \Phi), \Phi)\right]$ as before, where the expectation is now over both Φ and the internal randomness of π which determines $E(\pi, \Phi)$ for each Φ. We prove the following generalization of Theorem 5.

Theorem 32 *Fix any $\alpha \ge 1$ and item costs $c : E \to \mathbb{N}$. If f is adaptive monotone and adaptive submodular with respect to the distribution $\mathbb{P}[\Phi]$, and π is an α-approximate greedy policy, then for all policies π^* and positive integers ℓ and k*

$$f_{avg}(\pi_{[\ell]}) > \left(1 - e^{-\ell/\alpha k}\right) f_{avg}(\pi^*_{[k]}).$$

Proof The proof goes along the lines of the performance analysis of the greedy algorithm for maximizing a submodular function subject to a cardinality constraint of Nemhauser et al. (1978). An extension of that analysis to α-approximate greedy algorithms, which is analogous to ours but for the nonadaptive case, is shown by Goundan and Schulz (2007). For brevity, we will assume without loss of generality that $\pi = \pi_{[\ell]}$ and $\pi^* = \pi^*_{[k]}$. Then for all i, $0 \le i < \ell$

$$f_{\text{avg}}(\pi^*) \;\le\; f_{\text{avg}}(\pi_{[i]} @ \pi^*) \;\le\; f_{\text{avg}}(\pi_{[i]}) + \alpha k \left(f_{\text{avg}}(\pi_{[i+1]}) - f_{\text{avg}}(\pi_{[i]})\right) \tag{28}$$

The first inequality is due to the adaptive monotonicity of f and Lemma 30, from which we may infer $f_{\text{avg}}(\pi_2) \leq f_{\text{avg}}(\pi_1 @ \pi_2)$ for any π_1 and π_2. The second inequality may be obtained as a corollary of Lemma 31 as follows. Define a random partial realization $\Psi = \Psi(\Phi) := \left\{(e, \Phi(e)) : e \in E(\pi_{[i]}, \Phi)\right\}$, where Φ is distributed as $\mathbb{P}[\Phi]$. Consider $\Delta(\pi^* \mid \Psi)$, which equals the expected marginal benefit of the π^* portion of $\pi_{[i]} @ \pi^*$ conditioned on $\Phi \sim \Psi$. Lemma 31 allows us to bound it as

$$\mathbb{E}[\Delta(\pi^* \mid \Psi)] \leq \mathbb{E}[c(\pi^* \mid \Psi)] \cdot \max_e (\Delta(e \mid \Psi) / c(e)),$$

where the expectations are taken over the internal randomness of π^*, if there is any. Note that since π^* has the form $\pi'_{[k]}$ for some π' we know that for all Φ, $\mathbb{E}[c(E(\pi^*, \Phi))] \leq k$, where the expectation is again taken over the internal randomness of π^*. Hence $\mathbb{E}[c(\pi^* \mid \Psi)] \leq k$ for all Ψ. It follows that $\mathbb{E}[\Delta(\pi^* \mid \Psi)] \leq k \cdot \max_e (\Delta(e \mid \Psi) / c(e))$. By definition of an α-approximate greedy policy, π obtains at least $(1/\alpha) \max_e (\Delta(e \mid \Psi) / c(e)) \geq \mathbb{E}[\Delta(\pi^* \mid \Psi)] / \alpha k$ expected marginal benefit per unit cost in this case. Next we remove the conditioning on Ψ by taking expectations. For a random variable $X = X(\Psi)$, we let $\mathbb{E}_{\Psi \sim \mathbb{P}}[X]$ denote the expectation of X with respect to measure $\mathbb{P}[\Psi]$. Then

$$f_{\text{avg}}(\pi_{[i+1]}) - f_{\text{avg}}(\pi_{[i]}) \geq \mathbb{E}_{\Psi \sim \mathbb{P}}\left[\frac{1}{\alpha} \max_e \left(\frac{\Delta(e \mid \Psi)}{c(e)}\right)\right] \geq \mathbb{E}_{\Psi \sim \mathbb{P}}\left[\frac{\mathbb{E}[\Delta(\pi^* \mid \Psi)]}{\alpha k}\right] = \frac{f_{\text{avg}}(\pi_{[i]} @ \pi^*) - f_{\text{avg}}(\pi_{[i]})}{\alpha k}$$

which may be rearranged to yield the second inequality in (28).

Now define $\Delta_i := f_{\text{avg}}(\pi^*) - f_{\text{avg}}(\pi_{[i]})$, so that (28) implies $\Delta_i \leq \alpha k(\Delta_i - \Delta_{i+1})$, from which we infer $\Delta_{i+1} \leq \left(1 - \frac{1}{\alpha k}\right) \Delta_i$ and hence $\Delta_\ell \leq \left(1 - \frac{1}{\alpha k}\right)^\ell \Delta_0 < e^{-\ell/\alpha k} \Delta_0$, where for this last inequality we have used the fact that $1 - x < e^{-x}$ for all $x > 0$. Thus $f_{\text{avg}}(\pi^*) - f_{\text{avg}}(\pi_{[\ell]}) < e^{-\ell/\alpha k}\left(f_{\text{avg}}(\pi^*) - f_{\text{avg}}(\pi_{[0]})\right) \leq e^{-\ell/\alpha k} f_{\text{avg}}(\pi^*)$ so $f_{\text{avg}}(\pi) > (1 - e^{-\ell/\alpha k}) f_{\text{avg}}(\pi^*)$.

■

A.4 The Min-Cost-Cover Objective

In this section, we provide arbitrary item cost generalizations of Theorem 10 and Theorem 11. With item costs the Adaptive Stochastic Minimum Cost Cover problem becomes one of finding, for some quota on utility Q,

$$\pi^* \in \arg\min_\pi c_{\text{avg}}(\pi) \text{ such that } f(E(\pi, \Phi), \Phi) \geq Q \text{ for all } \Phi, \tag{29}$$

where $c_{\text{avg}}(\pi) := \mathbb{E}[c(E(\pi, \Phi))]$. Without loss of generality, we may take a truncated version of f, namely $(A, \Phi) \mapsto \min\{Q, f(A, \Phi)\}$, and rephrase Problem (29) as finding

$$\pi^* \in \arg\min_\pi c_{\text{avg}}(\pi) \text{ such that } \pi \text{ covers } \Phi \text{ for all } \Phi. \tag{30}$$

Hereby, recall that π covers Φ if $\mathbb{E}[f(E(\pi, \Phi), \Phi)] = f(E, \Phi)$, where the expectation is over any internal randomness of π. We will consider only Problem (30) for the remainder. We also consider the worst-case variant of this problem, where we replace the expected cost $c_{\text{avg}}(\pi)$ objective with the worst-case cost $c_{\text{wc}}(\pi) := \max_\Phi c(E(\pi, \Phi))$.

The definition of coverage (Definition 7 in §5.2 on page 10) requires no modification to handle item costs. Note, however, that coverage is all-or-nothing in the sense that covering a realization Φ with probability less than one does not count as covering it. A corollary of this is that only items whose runs have finished help with coverage, whereas currently running items do not. For a simple example, consider the case where $E = \{e\}$, $c(e) = 2$, $f(A, \Phi) = |A|$, and policy π that selects e and then terminates. Then $\pi_{[1]}$ is a randomized policy which is π with probability $\frac{1}{2}$, and is the empty policy with probability $\frac{1}{2}$, so $\mathbb{E}[f(E(\pi, \Phi), \Phi)] = \frac{1}{2} < 1 = f(E, \Phi)$ for each Φ. Hence, even though half the time $\pi_{[1]}$ covers all realizations, it is counted as not covering any.

We begin with the approximation guarantee for the average-case policy cost with arbitrary item costs.

Theorem 33 *Suppose $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$ is adaptive submodular and strongly adaptive monotone with respect to $\mathbb{P}[\Phi]$ and there exists Q such that $f(E, \Phi) = Q$ for all Φ. Let η be any value such that $f(S, \Phi) > Q - \eta$ implies $f(S, \Phi) = Q$ for all S and Φ. Let $\delta = \min_\Phi \mathbb{P}[\Phi]$ be the minimum probability of any realization. Let π^*_{avg} be an optimal policy minimizing the expected number of items selected to guarantee every realization is covered. Let π be an α-approximate greedy policy with respect to the item costs. Then in general*

$$c_{avg}(\pi) \leq \alpha\, c_{avg}(\pi^*_{avg}) \left(\ln\left(\frac{Q}{\delta\eta}\right) + 1 \right)$$

and for self–certifying instances

$$c_{avg}(\pi) \leq \alpha\, c_{avg}(\pi^*_{avg}) \left(\ln\left(\frac{Q}{\eta}\right) + 1 \right).$$

Note that if $\operatorname{range}(f) \subset \mathbb{Z}$, then $\eta = 1$ is a valid choice, so in this case $c_{avg}(\pi) \leq \alpha\, c_{avg}(\pi^*_{avg}) (\ln(Q/\delta) + 1)$ and $c_{avg}(\pi) \leq \alpha\, c_{avg}(\pi^*_{avg}) (\ln(Q) + 1)$ for general and self–certifying instances, respectively.

Proof Consider running α-approximate greedy policy π to completion, i.e., until it covers the true realization. It starts off with $v_0 := \mathbb{E}[f(\emptyset, \Phi)] \geq 0$ reward in expectation, and terminates with Q reward. Along the way it will go through some sequence of partial realizations specifying its current observations, $\Psi_0 \subset \Psi_1 \subset \cdots \subset \Psi_\ell$, such that $\operatorname{dom}(\Psi_i) \setminus \operatorname{dom}(\Psi_{i-1})$ consists precisely of the i^{th} item selected by π. We call this sequence the *trace* $\tau = \tau(\Phi)$ of π. For a realization Φ and $x \in \mathbb{R}_{\geq 0}$, we define $\Psi(\Phi, x)$ as the partial realization seen by π just before it achieved x reward in expectation. Formally,

$$\Psi(\Phi, x) \in \arg\max \{ |\operatorname{dom}(\Psi)| \;:\; \Psi \in \tau(\Phi),\ \mathbb{E}[f(\operatorname{dom}(\Psi), \Phi) \mid \Phi \sim \Psi] < x \} \tag{31}$$

Note that $\Psi(\Phi, x)$ exists for all $x \in (v_0, Q]$, and when it exists it is unique since no two elements of the trace have equally large domains. Also note that by the strong adaptive monotonicity of f, the function $i \mapsto \mathbb{E}[f(\operatorname{dom}(\Psi_i), \Phi) \mid \Phi \sim \Psi_i]$ must be nondecreasing for any trace $\Psi_0, \Psi_1, \ldots, \Psi_\ell$.

Our overall strategy will be to bound the expected cost $c_{\text{avg}}(\pi)$ of π by bounding the price it pays per unit of *expected reward* gained as it runs, and then integrating over the run. Note that Lemma 31 tells us that $\max_e (\Delta(e \,|\, \Psi) / c(e)) \geq \Delta\left(\pi^*_{\text{avg}} \,|\, \Psi\right) / c\left(\pi^*_{\text{avg}} \,|\, \Psi\right)$ for all Ψ. An α-approximate greedy policy obtains at least $1/\alpha$ of this rate. Hence we may bound its price, θ, by $\theta(\Psi) \leq \alpha\, c\left(\pi^*_{\text{avg}} \,|\, \Psi\right) / \Delta\left(\pi^*_{\text{avg}} \,|\, \Psi\right)$.

Rather than try to bound the expected price as π progresses in time, we will bound the expected price as it progresses in the expected reward it obtains, measured as $\mathbb{E}[f(\operatorname{dom}(\Psi), \Phi) \mid \Phi \sim \Psi]$ where Ψ is the current partial realization. We next claim that $\Delta\left(\pi^*_{\text{avg}} \,|\, \Psi(\Phi, x)\right) \geq Q - x$ for all Φ and x. Note that $\mathbb{E}[f(\operatorname{dom}(\Psi(\Phi, x)), \Phi) \mid \Phi \sim \Psi(\Phi, x)] < x$ by definition of $\Psi(\Phi, x)$, and $f(E(\pi^*_{\text{avg}}, \Phi), \Phi) = Q$ for all Φ since π^*_{avg} covers every realization. Since Q is the maximum possible reward, if $\Delta\left(\pi^*_{\text{avg}} \,|\, \Psi(\Phi, x)\right) < Q - x$ then we can generate a violation of strong adaptive monotonicity by fixing some $\Phi \sim \Psi(\Phi, x)$, selecting $E(\pi^*_{\text{avg}}, \Phi)$, and then selecting $\operatorname{dom}(\Psi(\Phi, x))$ to reduce the expected reward. Thus $\Delta\left(\pi^*_{\text{avg}} \,|\, \Psi(\Phi, x)\right) \geq Q - x$, and we infer

$$\theta(\Psi(\Phi, x)) \;\leq\; \frac{\alpha\, c\left(\pi^*_{\text{avg}} \,|\, \Psi(\Phi, x)\right)}{\Delta\left(\pi^*_{\text{avg}} \,|\, \Psi(\Phi, x)\right)} \;\leq\; \frac{\alpha\, c\left(\pi^*_{\text{avg}} \,|\, \Psi(\Phi, x)\right)}{Q - x} \tag{32}$$

Next, we take an expectation over Φ. Let $\theta(x) := \mathbb{E}[\theta(\Psi(\Phi, x))]$. Let $\Psi_1^x, \ldots, \Psi_r^x$ be the possible values of $\Psi(\Phi, x)$. Then because $\{\{\Phi : \Phi \sim \Psi_i^x\} : i = 1, 2, \ldots, r\}$ partitions the set of realizations,

$$\begin{aligned}
\mathbb{E}\left[c\left(\pi^*_{\text{avg}} \,|\, \Psi(\Phi, x)\right)\right] &= \sum_{i=1}^{r} \mathbb{P}[\Psi_i^x] \sum_{\Phi} \mathbb{P}[\Phi \mid \Psi_i^x] \cdot c\left(\pi^*_{\text{avg}} \,|\, \Phi\right) && (33)\\
&= \sum_{\Phi} \mathbb{P}[\Phi] \cdot c\left(\pi^*_{\text{avg}} \,|\, \Phi\right) && (34)\\
&= c_{\text{avg}}(\pi^*_{\text{avg}}) && (35)
\end{aligned}$$

It follows that

$$\theta(x) \leq \frac{\alpha\, c_{\text{avg}}(\pi^*_{\text{avg}})}{Q - x}. \tag{36}$$

Let $c_{\text{avg}}(\pi, Q')$ denote the expected cost to obtain expected reward Q'. Then we can bound $c_{\text{avg}}(\pi, Q')$ as

$$c_{\text{avg}}(\pi, Q') \;=\; \int_{x=0}^{Q'} \theta(x) dx \;\leq\; \int_{x=0}^{Q'} \frac{\alpha\, c_{\text{avg}}(\pi^*)}{Q - x} dx \;=\; \alpha\, c_{\text{avg}}(\pi^*) \ln\left(\frac{Q}{Q - Q'}\right) \tag{37}$$

We now use slightly different analyses for general instances and for self–certifying instances. We begin with general instances. For these, we set $Q' = Q - \delta\eta$ and use a more refined argument to bound the cost of getting the remaining expected reward. Fix $\Psi \in \text{dom}(\pi)$ and any $\Phi' \sim \Psi$. We say Ψ *covers* Φ' if π covers Φ' by the time it observes Ψ. By definition of δ and η, if some Φ' is not covered by Ψ then $Q - \mathbb{E}\left[f(\text{dom}(\Psi), \Phi) \mid \Phi \sim \Psi\right] \geq \delta\eta$. Hence the last item that π selects, say upon observing Ψ, must increase its conditional expected value from $\mathbb{E}\left[f(\text{dom}(\Psi), \Phi) \mid \Phi \sim \Psi\right] \leq Q - \delta\eta$ to Q. By Eq. (32), it follows that for $x \in [Q - \delta\eta, Q]$,

$$\theta(\Psi\left(\Phi, x\right)) \;\leq\; \frac{\alpha\, c\left(\pi^*_{\text{avg}} \,|\, \Psi\left(\Phi, x\right)\right)}{\Delta\left(\pi^*_{\text{avg}} \,|\, \Psi\left(\Phi, x\right)\right)} \;\leq\; \frac{\alpha\, c\left(\pi^*_{\text{avg}} \,|\, \Psi\left(\Phi, x\right)\right)}{\delta\eta}.$$

As before, we may take the expectation over Φ to obtain $\theta(x) \leq \alpha\, c_{\text{avg}}(\pi^*_{\text{avg}})/\delta\eta$ for all $x \in [Q - \delta\eta, Q]$. This fact together with Eq. (37) yield

$$\begin{aligned} c_{\text{avg}}(\pi) \;\equiv\; c_{\text{avg}}(\pi, Q) &\;=\; c_{\text{avg}}(\pi, Q - \delta\eta) + \textstyle\int_{x=Q-\delta\eta}^{Q} \mathbb{E}\left[\theta(x)\right] dx \\ &\;\leq\; \alpha\, c_{\text{avg}}(\pi^*) \ln\left(Q/\delta\eta\right) + \textstyle\int_{x=Q-\delta\eta}^{Q} \frac{\alpha c_{\text{avg}}(\pi^*)}{\delta\eta} dx \\ &\;=\; \alpha\, c_{\text{avg}}(\pi^*) \left(\ln\left(Q/\delta\eta\right) + 1\right) \end{aligned}$$

which completes the proof for general instances.

For self–certifying instances we use a similar argument. For these instances we set $Q' = Q - \eta$, and argue that the last item that π selects must increase its conditional expected value from at most $Q - \eta$ to Q. For suppose π currently observes Ψ, and has not achieved conditional value Q, i.e., $\mathbb{E}\left[f(\text{dom}(\Psi), \Phi) \mid \Phi \sim \Psi\right] < Q$. Then some $\Phi \sim \Psi$ is uncovered. Since the instance is self–certifying, every Φ with $\Phi \sim \Psi$ then has $f(\text{dom}(\Psi), \Phi) < f(E, \Phi) = Q$. By definition of η, for each Φ with $\Phi \sim \Psi$ we then have $f(\text{dom}(\Psi), \Phi) \leq Q - \eta$, which implies $\mathbb{E}\left[f(\text{dom}(\Psi), \Phi) \mid \Phi \sim \Psi\right] \leq Q - \eta$. Reasoning analogously as with general instances, we may derive from this that $\theta(x) \leq \alpha\, c_{\text{avg}}(\pi^*_{\text{avg}})/\eta$ for all $x \in [Q - \eta, Q]$. Computing $c_{\text{avg}}(\pi) = c_{\text{avg}}(\pi, Q') + \int_{x=Q'}^{Q} \mathbb{E}\left[\theta(x)\right] dx$ as before gives us the claimed approximation ratio for self–certifying instances, and completes the proof. ■

Next we consider the worst-case cost. We generalize Theorem 11 by incorporating arbitrary item costs.

Theorem 34 *Suppose $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$ is adaptive monotone and adaptive submodular with respect to $\mathbb{P}\left[\Phi\right]$, and let η be any value such that $f(S, \Phi) > f(E, \Phi) - \eta$ implies $f(S, \Phi) = f(E, \Phi)$ for all S and Φ. Let $\delta = \min_\Phi \mathbb{P}\left[\Phi\right]$ be the minimum probability of any realization. Let π^*_{wc} be the optimal policy minimizing the worst-case cost $c_{wc}(\cdot)$ while guaranteeing that every realization is covered. Let π be an α-approximate greedy policy with respect to the item costs. Finally, let $Q := \mathbb{E}\left[f(E, \Phi)\right]$ be the maximum possible expected reward. Then*

$$c_{wc}(\pi) \leq \alpha\, c_{wc}(\pi^*_{wc}) \left(\ln\left(\frac{Q}{\delta\eta}\right) + 1\right).$$

Proof Let π be an α-approximate greedy policy. Let $k = c_{\text{wc}}(\pi^*_{wc})$, let $\ell = \alpha k \ln\left(Q/\delta\eta\right)$, and apply Theorem 32 with these parameters to yield

$$f_{\text{avg}}(\pi_{[\ell]}) \;>\; \left(1 - e^{-\ell/\alpha k}\right) f_{\text{avg}}(\pi^*_{wc}) \;=\; \left(1 - \frac{\delta\eta}{Q}\right) f_{\text{avg}}(\pi^*_{wc}). \tag{38}$$

Since π^*_{wc} covers every realization by assumption, $f_{\text{avg}}(\pi^*_{wc}) = \mathbb{E}[f(E, \Phi)] = Q$, so rearranging terms of Eq. (38) yields $Q - f_{\text{avg}}(\pi_{[\ell]}) < \delta\eta$. Since $f_{\text{avg}}(\pi_{[\ell]}) \leq f_{\text{avg}}(\pi_{[\ell\rightarrow]})$ by the adaptive monotonicity of f, it follows that $Q - f_{\text{avg}}(\pi_{[\ell\rightarrow]}) < \delta\eta$. By definition of δ and η, if some Φ is not covered by $\pi_{[\ell\rightarrow]}$ then $Q - f_{\text{avg}}(\pi_{[\ell\rightarrow]}) \geq \delta\eta$. Thus $Q - f_{\text{avg}}(\pi_{[\ell\rightarrow]}) < \delta\eta$ implies $Q - f_{\text{avg}}(\pi_{[\ell\rightarrow]}) = 0$, meaning $\pi_{[\ell\rightarrow]}$ covers every realization.

We next claim that $\pi_{[\ell\rightarrow]}$ has worst-case cost at most $\ell + \alpha k$. It is sufficient to show that the final item executed by $\pi_{[\ell\rightarrow]}$ has cost at most αk for any realization. As we will prove, this follows from the facts that π is an α-approximate greedy policy and π^*_{wc} covers every realization at cost at most k. The data dependent bound, Lemma 31 on page 36, guarantees that

$$\max_e \left(\frac{\Delta(e \,|\, \Psi)}{c(e)} \right) \geq \frac{\Delta(\pi^*_{wc} \,|\, \Psi)}{c(\pi^*_{wc} \,|\, \Psi)} \geq \frac{\Delta(\pi^*_{wc} \,|\, \Psi)}{k} \tag{39}$$

Suppose $\Psi \in \text{dom}(\pi)$. We would like to say that $\max_e \Delta(e \,|\, \Psi) \leq \Delta(\pi^*_{wc} \,|\, \Psi)$. Supposing this is true, any item e with cost $c(e) > \alpha k$ must have $\Delta(e \,|\, \Psi) / c(e) < \Delta(\pi^*_{wc} \,|\, \Psi) / \alpha k$, and hence cannot be selected by any α-approximate greedy policy upon observing Ψ by Eq. (39), and thus the final item executed by $\pi_{[\ell\rightarrow]}$ has cost at most αk for any realization. So we next show that $\max_e \Delta(e \,|\, \Psi) \leq \Delta(\pi^*_{wc} \,|\, \Psi)$. Towards this end, note that Lemma 35 implies

$$\max_e \Delta(e \,|\, \Psi) \leq \mathbb{E}[f(E, \Phi) \mid \Phi \sim \Psi] - \mathbb{E}[f(\text{dom}(\Psi), \Phi) \mid \Phi \sim \Psi]. \tag{40}$$

and to prove $\max_e \Delta(e \,|\, \Psi) \leq \Delta(\pi^*_{wc} \,|\, \Psi)$ it suffices to show

$$\mathbb{E}[f(E, \Phi) \mid \Phi \sim \Psi] \leq \mathbb{E}[f(E(\pi^*_{wc}, \Phi) \cup \text{dom}(\Psi), \Phi) \mid \Phi \sim \Psi]. \tag{41}$$

Proving Eq. (41) is quite straightforward if f is *strongly* adaptive monotone. Given that f is only adaptive monotone, it requires some additional effort. So fix $A \subset E$ and let π_A be a non-adaptive policy that selects all items in A in some arbitrary order. Let $\mathcal{P} := \{\Psi : \text{dom}(\Psi) = A\}$. Apply Lemma 35 with $\pi' = \pi_A @ \pi^*_{wc}$ and any $\Psi \in \mathcal{P}$ to obtain

$$\mathbb{E}[f(E(\pi^*_{wc}, \Phi) \cup A, \Phi) \mid \Phi \sim \Psi] \leq \mathbb{E}[f(E, \Phi) \mid \Phi \sim \Psi]. \tag{42}$$

Note that $\sum_{\Psi \in \mathcal{P}} \mathbb{P}[\Psi] \cdot \mathbb{E}[f(E(\pi^*_{wc}, \Phi) \cup A, \Phi) \mid \Phi \sim \Psi] = f_{\text{avg}}(\pi_A @ \pi^*_{wc}) \geq f_{\text{avg}}(\pi^*_{wc}) = \mathbb{E}[f(E, \Phi)]$. Since we know $\mathbb{E}[f(E, \Phi)] = \sum_{\Psi \in \mathcal{P}} \mathbb{P}[\Psi] \, \mathbb{E}[f(E, \Phi) \mid \Phi \sim \Psi]$, an averaging argument together with Eq. (42) then implies that for all $\Psi \in \mathcal{P}$

$$\mathbb{E}[f(E(\pi^*_{wc}, \Phi) \cup A, \Phi) \mid \Phi \sim \Psi] = \mathbb{E}[f(E, \Phi) \mid \Phi \sim \Psi] \tag{43}$$

Since Ψ was an arbitrary partial realization with $\text{dom}(\Psi) = A$, and $A \subseteq E$ was arbitrary, fix $\Psi \in \text{dom}(\pi)$ and let $A = \text{dom}(\Psi)$. With these settings, Eq. (43) implies Eq. (41), and thus $\max_e \Delta(e \,|\, \Psi) \leq \Delta(\pi^*_{wc} \,|\, \Psi)$, and thus an α-approximate greedy policy can never select an item with cost exceeding αk, where $k = c_{\text{wc}}(\pi^*_{wc})$. Hence $c_{\text{wc}}(\pi_{[\ell\rightarrow]}) - c_{\text{wc}}(\pi_{[\ell]}) \leq \alpha k$, and so $c_{\text{wc}}(\pi_{[\ell\rightarrow]}) \leq \ell + \alpha k$. This completes the proof. ■

Lemma 35 *Fix adaptive monotone submodular objective f. For any policy π and any $\Psi \in \text{dom}(\pi)$ we have*

$$\mathbb{E}[f(E(\pi, \Phi), \Phi) \mid \Phi \sim \Psi] \leq \mathbb{E}[f(E, \Phi) \mid \Phi \sim \Psi].$$

Proof Augment π to a new policy π' as follows. Run π to completion, and let Ψ' be the partial realization consisting of all of the states it has observed. If $\Psi \subseteq \Psi'$, then proceed to select all the remaining items in E in any order. Otherwise, if $\Psi \not\subseteq \Psi'$ then terminate. Then

$$\mathbb{E}[f(E(\pi, \Phi), \Phi) \mid \Phi \sim \Psi] \leq \mathbb{E}[f(E(\pi', \Phi), \Phi) \mid \Phi \sim \Psi] = \mathbb{E}[f(E, \Phi) \mid \Phi \sim \Psi] \tag{44}$$

where the inequality is by repeated application of the adaptive monotonicity of f, and the equality is by construction. ■

In §5.2 we explained how the result of Feige (1998) implies there is no polynomial time $(1-\epsilon)\ln(Q/\eta)$ approximation algorithm for self–certifying instances of Adaptive Stochastic Min Cost Cover, unless NP $\subseteq$ DTIME($n^{\mathcal{O}(\log\log n)}$). Here we show the related result for general instances.

Lemma 36 *For every constant $\epsilon > 0$, there is no $(1-\epsilon)\ln(Q/\delta\eta)$ polynomial time approximation algorithm for general instances of Adaptive Stochastic Min Cost Cover, for either the average case objective $c_{avg}(\cdot)$ or the worst-case objective $c_{wc}(\cdot)$, unless NP $\subseteq$ DTIME($n^{\mathcal{O}(\log\log n)}$).*

Proof We offer a reduction from the Set Cover problem. Fix a Set Cover instance $U, \{S_1, S_2, \ldots, S_m\} \subseteq 2^U$ with unit-cost sets. Fix Q, η and δ such that $1/\delta$ and Q/η are positive integers, and $\frac{Q}{\delta\eta} = |U|$. Let $E := \{S_1, S_2, \ldots, S_m\}$, and set of costs of all items to one. Partition U into $1/\delta$ disjoint, equally sized subsets $U_1, U_2, \ldots, U_{1/\delta}$. Construct a realization Φ_i for each U_i. Let the set of states be $O = \{\text{NULL}\}$. Hence $\Phi_i(e) = \text{NULL}$ for all i and e, so that no knowledge of the true realization is revealed by selecting items. We use a uniform distribution over realizations, i.e., $\mathbb{P}[\Phi_i] = \delta$ for all i. Finally, our objective is $f(\mathcal{C}, \Phi_i) := |\cup_{S\in\mathcal{C}} (S \cap U_i)|$, i.e., the number of elements in U_i that we cover with sets in $\mathcal{C}$. Since $|O| = 1$, every realization is consistent with every possible partial realization Ψ. Hence for any Ψ, we have $\mathbb{E}[f(\mathrm{dom}(\Psi), \Phi) \mid \Phi \sim \Psi] = \delta \hat{f}(\mathrm{dom}(\Psi))$, where $\hat{f}(\mathcal{C}) = |\cup_{S\in\mathcal{C}} S|$ is the objective function of the original set cover instance. Since $\hat{f}$ is submodular, f is adaptive submodular. Likewise, since $\hat{f}$ is monotone, and $|O| = 1$, f is strongly adaptive monotone. Now, to cover any realization, we must obtain the maximum possible value for all realizations, which means selecting a collection of sets $\mathcal{C}$ such that $\cup_{S\in\mathcal{C}} S = U$. Conversely, any $\mathcal{C}$ such that $\cup_{S\in\mathcal{C}} S = U$ clearly covers f. Hence this instance of Adaptive Stochastic Min Cost Cover, with either the average case objective $c_{\text{avg}}(\cdot)$ or the worst-case objective $c_{\text{wc}}(\cdot)$, is equivalent to the original Set Cover instance. Therefore, the result from Feige (1998) implies that there is no polynomial time algorithm for obtaining a $(1-\epsilon)\ln|U| = (1-\epsilon)\ln(Q/\delta\eta)$ approximation for Adaptive Stochastic Min Cost Cover unless NP $\subseteq$ DTIME($n^{\mathcal{O}(\log\log n)}$). ■

A.5 The Min-Sum Objective

In this section we prove Theorem 12, which appears on page 13, in the case where the items have arbitrary costs. Our proof resembles the analogous proof of Streeter and Golovin (2007) for the non-adaptive min-sum submodular cover problem, and, like that proof, ultimately derives from an extremely elegant performance analysis of the greedy algorithm for min-sum set cover due to Feige et al. (2004).

The objective function $c_\Sigma(\cdot)$ generalized to arbitrary cost items uses $\pi_{[\leftarrow t]}$ in place of $\pi_{[t]}$ in the unit-cost definition:

$$c_\Sigma(\pi) := \sum_{t=0}^{\infty} \left(\mathbb{E}[f(E,\Phi)] - f_{\text{avg}}(\pi_{[\leftarrow t]})\right) = \sum_{\Phi} \mathbb{P}[\Phi] \sum_{t=0}^{\infty} \left(f(E,\Phi) - f(E(\pi_{[\leftarrow t]},\Phi),\Phi)\right) \tag{45}$$

We will prove that any α-approximate greedy policy π achieves a 4α-approximation for the min-sum objective, i.e., $c_\Sigma(\pi) \leq 4\alpha\, c_\Sigma(\pi^*)$ for all policies π^*. To do so, we require the following lemma.

Lemma 37 *Fix an α-approximate greedy policy π for some adaptive monotone submodular function f and let $s_i := \alpha\left(f_{avg}(\pi_{[i+1]}) - f_{avg}(\pi_{[i]})\right)$. For any policy π^* and nonnegative integers i and k, we have $f_{avg}(\pi^*_{[k]}) \leq f_{avg}(\pi_{[\leftarrow i]}) + k \cdot s_i$.*

Proof Fix π, π^*, i, and k. By adaptive monotonicity $f_{\text{avg}}(\pi^*_{[k]}) \leq f_{\text{avg}}(\pi_{[\leftarrow i]} @ \pi^*_{[k]})$. We next aim to prove

$$f_{\text{avg}}(\pi_{[\leftarrow i]} @ \pi^*_{[k]}) \leq f_{\text{avg}}(\pi_{[\leftarrow i]}) + k \cdot s_i \tag{46}$$

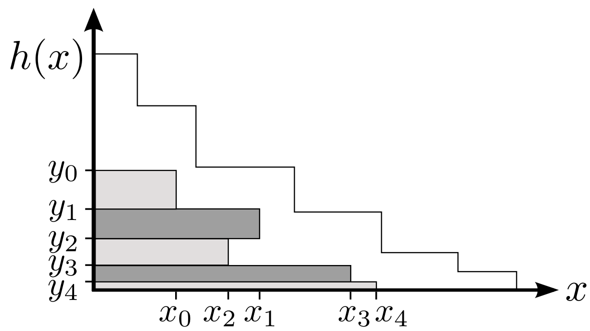


Figure 6: An illustration of the inequality $\int_{x=0}^{\infty} h(x)dx \geq \sum_{i\geq 0} x_i\,(y_i - y_{i+1})$.

which is sufficient to complete the proof. Towards this end, define a random partial realization $\Psi = \Psi(\Phi) := \{(e, \Phi(e)) : e \in E(\pi_{[\leftarrow i]}, \Phi)\}$, where Φ is distributed as $\mathbb{P}[\Phi]$. Consider the expected marginal benefit of the $\pi^*_{[k]}$ portion of $\pi_{[\leftarrow i]}@\pi^*_{[k]}$ conditioned on $\Phi \sim \Psi$. Consider $\Delta\left(\pi^*_{[k]} \,|\, \Psi\right)$, which equals the expected marginal benefit of the $\pi^*_{[k]}$ portion of $\pi_{[\leftarrow i]}@\pi^*_{[k]}$ conditioned on $\Phi \sim \Psi$. Lemma 31 allows us to bound it as

$$\mathbb{E}\left[\Delta\left(\pi^*_{[k]} \,|\, \Psi\right)\right] \;\leq\; \mathbb{E}\left[c\left(\pi^*_{[k]} \,|\, \Psi\right)\right] \cdot \max_e \left(\frac{\Delta(e \,|\, \Psi)}{c(e)}\right),$$

where the expectations are taken over the internal randomness of π^*, if there is any. Note that for all Φ, $\mathbb{E}\left[c(E(\pi^*_{[k]}, \Phi))\right] \leq k$, where the expectation is again taken over the internal randomness of $\pi^*_{[k]}$. Hence $\mathbb{E}\left[c\left(\pi^*_{[k]} \,|\, \Psi\right)\right] \leq k$ for all Ψ. It follows that $\mathbb{E}\left[\Delta\left(\pi^*_{[k]} \,|\, \Psi\right)\right] \leq k \cdot \max_e\left(\Delta(e \,|\, \Psi)/c(e)\right)$. By definition of an α-approximate greedy policy, π obtains at least $(1/\alpha)\max_e\left(\Delta(e \,|\, \Psi)/c(e)\right) \geq \mathbb{E}\left[\Delta\left(\pi^*_{[k]} \,|\, \Psi\right)\right]/\alpha k$ benefit per unit cost this case. Hence, removing the conditioning on Ψ by taking expectations

$$f_{\text{avg}}(\pi_{[i+1]}) - f_{\text{avg}}(\pi_{[i]}) \geq \mathbb{E}_\Psi\left[\frac{\mathbb{E}\left[\Delta\left(\pi^*_{[k]} \,|\, \Psi\right)\right]}{\alpha k}\right] = \frac{f_{\text{avg}}(\pi_{[\leftarrow i]}@\pi^*_{[k]}) - f_{\text{avg}}(\pi_{[\leftarrow i]})}{\alpha k}.$$

Multiplying the above inequalities by αk, and substituting in $s_i = \alpha\left(f_{\text{avg}}(\pi_{[i+1]}) - f_{\text{avg}}(\pi_{[i]})\right)$, we conclude $ks_i \geq f_{\text{avg}}(\pi_{[\leftarrow i]}@\pi^*_{[k]}) - f_{\text{avg}}(\pi_{[\leftarrow i]})$ which immediately yields Eq. (46) and concludes the proof. ■

Using Lemma 37, together with a geometric argument developed by Feige et al. (2004), we now prove Theorem 12.

Proof of Theorem 12: Let $Q := \mathbb{E}[f(E, \Phi)]$ be the maximum possible expected reward, where the expectation is taken w.r.t. $\mathbb{P}[\Phi]$. Let π be an α-approximate greedy policy. Define $R_i := Q - f_{\text{avg}}\left(\pi_{[i]}\right)$ and define $P_i := Q - f_{\text{avg}}\left(\pi_{[\leftarrow i]}\right)$. Let $x_i := \frac{P_i}{2s_i}$, let $y_i := \frac{R_i}{2}$, and let $h(x) := Q - f_{\text{avg}}(\pi^*_{[x]})$. We claim $f_{\text{avg}}\left(\pi_{[\leftarrow i]}\right) \leq f_{\text{avg}}\left(\pi_{[i]}\right)$ and so $P_i \geq R_i$. This clearly holds if $\pi_{[\leftarrow i]}$ is the empty policy, and otherwise π can always select an item that contributes zero marginal benefit, namely an item it has already played previously. Hence an α-approximate greedy policy π can never select items with negative expected marginal benefit, and so $f_{\text{avg}}\left(\pi_{[\leftarrow i]}\right) \leq f_{\text{avg}}\left(\pi_{[i]}\right)$. By Lemma 37, $f_{\text{avg}}\left(\pi^*_{[x_i]}\right) \leq f_{\text{avg}}\left(\pi_{[\leftarrow i]}\right) + x_i s_i$. Therefore

$$h(x_i) \geq Q - f_{\text{avg}}(\pi_{[\leftarrow i]}) - x_i \cdot s_i = P_i - \frac{P_i}{2} \geq \frac{R_i}{2} = y_i \tag{47}$$

For similar reasons that $f_{\text{avg}}\left(\pi_{[\leftarrow i]}\right) \leq f_{\text{avg}}\left(\pi_{[i]}\right)$, we have $f_{\text{avg}}\left(\pi_{[i-1]}\right) \leq f_{\text{avg}}\left(\pi_{[i]}\right)$, and so the sequence $\langle y_1, y_2, \ldots \rangle$ is non-increasing. The adaptive monotonicity and adaptive submodularity of f imply that $h(x)$ is non-increasing. Informally, this is because otherwise, if $f_{\text{avg}}(\pi^*_{[x]}) > f_{\text{avg}}(\pi^*_{[x+1]})$ for some x, then the optimal policy must be sacrificing immediate rewards at time x in exchange for greater returns later, and it can be shown that if such a strategy is optimal, then adaptive submodularity cannot hold. Eq. (47) and the monotonicity of h and $i \mapsto y_i$ imply that $\int_{x=0}^{\infty} h(x)dx \geq \sum_{i\geq 0} x_i\left(y_i - y_{i+1}\right)$ (see Figure 6). The left hand side is a lower bound for $c_\Sigma(\pi^*)$, and because $s_i = \alpha\left(R_i - R_{i+1}\right)$ the right hand side simplifies to $\frac{1}{4\alpha}\sum_{i\geq 0} P_i = \frac{1}{4\alpha}c_\Sigma(\pi)$, proving $c_\Sigma(\pi) \leq 4\alpha \cdot c_\Sigma(\pi^*)$. ■

A.6 Proof of Approximation Hardness in the Absence of Adaptive Submodularity

We now provide the proof of Theorem 20 which appears on page 28 in §12.

Proof of Theorem 20: We construct a hard instance based on the following intuition. We make the algorithm go "treasure hunting". There is a set of t locations $\{0, 1, , \ldots, t-1\}$, there is a treasure at one of these locations, and the algorithm gets unit reward if it finds it, and zero reward otherwise. There are m "maps," each consisting of a cluster of s bits, and each purporting to indicate where the treasure is, and each map is stored in a (weak) secret-sharing way, so that querying few bits of a map reveals nothing about where it says the treasure is. Moreover, all but one of the maps are *fake*, and there is a puzzle indicating which map is the correct one indicating the treasure's location. Formally, a fake map is one which is probabilistically independent of the location of the treasure, conditioned on the puzzle.

Our instance will have three types of elements, $E = E_T \uplus E_M \uplus E_P$, where $|E_T| = t$ encodes where the treasure is, $|E_M| = ms$ encodes the maps, and $|E_P| = n^3$ encodes the puzzle, where m, t, s and n are specified below. All outcomes are binary, $O = \{0, 1\}$. For all $e \in E_M \cup E_P$, $\mathbb{P}\left[\Phi(e) = 1\right] = .5$ independently. The conditional distribution $\mathbb{P}\left[\Phi(E_T) \mid \Phi(E_M \cup E_P)\right]$ will be deterministic as specified below. Our objective function f is linear, and defined as follows:

$$f(A, \Phi) = |\{e \in A \cap E_T : \Phi(e) = 1\}|.$$

We now describe the puzzle, which is to compute $i(P) := (\text{perm}(P) \mod p) \mod 2^\ell$ for a suitably sampled random matrix P, and suitable prime p and integer ℓ, where $\text{perm}(P) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} P_{i\sigma(i)}$ is the permanent of P. We exploit Theorem 1.9 of Feige and Lund (1997) in which they show that if there exist constants $\eta, \delta > 0$ such that a randomized polynomial time algorithm can compute $(\text{perm}(P) \mod p) \mod 2^\ell$ correctly with probability $2^{-\ell}(1 + 1/n^\eta)$, where P is drawn uniformly at random from $\{0, 1, 2, \ldots, p-1\}^{n\times n}$, p is any prime superpolynomial in n, and $\ell \leq p\left(\frac{1}{2} - \delta\right)$, then PH = AM = Σ_2^P. To encode the puzzle, we fix a prime $p \in [2^{n-2}, 2^{n-1}]$ and use the n^3 bits of $\Phi(E_P)$ to sample $P = P(\Phi)$ (nearly) uniformly at random from $\{0, 1, 2, \ldots, p-1\}^{n\times n}$ as follows. For a matrix $P \in \mathbb{Z}^{n\times n}$, we let $\text{rep}(P) := \sum_{ij} P_{ij} \cdot p^{(i-1)n+(j-1)}$ define a base p representation of P. Note $\text{rep}(\cdot)$ is one-to-one for $n \times n$ matrices with entries in $\mathbb{Z}_p$, so we can define its inverse $\text{rep}^{-1}(\cdot)$. The encoding $P(\Phi)$ interprets the bits $\Phi(E_P)$ as an integer x in $[2^{n^3}]$, and computes $y = x \mod (p^{n^2})$. If $x \leq \left\lfloor 2^{n^3}/p^{n^2} \right\rfloor p^{n^2}$, then $P = \text{rep}^{-1}(y)$. Otherwise, P is the all zero matrix. This latter event occurs with probability at most $p^{n^2}/2^{n^3} \leq 2^{-n^2}$, and in this case we simply suppose the algorithm under consideration finds the treasure and so gets unit reward. This adds 2^{-n^2} to its expected reward. So let us assume from now on that P is drawn uniformly at random.

Next we consider the maps. Partition $E_M = \biguplus_{i=1}^{m} M_i$ into m maps M_i, each consisting of s items. For each map M_i, partition its items into $s/\log_2 t$ groups of $\log_2 t$ bits each, and let $v_i \in \{0, 1, \ldots, t-1\}$ be the XOR of these groups of bits. We say M_i *points to* v_i as the location of the treasure. A priori, each v_i is uniformly distributed in $\{0, ..., t-1\}$. For a particular realization of $\Phi(E_P \cup E_M)$, define $v(\Phi) := v_{i(P(\Phi))}$. We set $v(\Phi)$ to be the location of the treasure under realization Φ, i.e., we label $E_T = \{e_0, e_1, \ldots, e_{t-1}\}$ and ensure $\Phi(e_j) = 1$ if $j = v_{i(P(\Phi))}$, and $\Phi(e) = 0$ for all other $e \in E_T$. Note the random variable $v = v(\Phi)$ is distributed uniformly at random in $\{0, 1, \ldots, t-1\}$. Note that this still holds if we condition on the realizations of any set of $s/\log_2 t - 1$ items in a map.

Now consider the optimal policy with a budget of $k = n^3 + s + 1$ items to pick. Clearly, its reward can be at most 1. However, given a budget of k, a computationally unconstrained policy can exhaustively sample E_P, solve the puzzle (i.e., compute $i(P)$), read the correct map (i.e., exhaustively sample $M_{i(P)}$), decode the map (i.e., compute $v = v_{i(P)}$), and get the treasure (i.e., pick e_v) thereby obtaining a reward of one.

Now we give an upper bound on the expected reward R of any randomized polynomial time algorithm $\mathcal{A}$ with a budget of βk items, assuming $\Sigma_2^P \neq$ PH. Fix a small constant $\gamma > 0$, and set $s = n^3$ and $m = t = n^{1/\gamma}$. We suppose we give $\mathcal{A}$ the realizations $\Phi(E_M)$ for free. We also replace its budget of βk items with a budget of βk specifically for map items in E_M and an additional budget of βk specifically for the treasure locations in E_T. Obviously, this can only help it. As noted, if it selects less than $s/\log_2 t$ bits from the map $M_{i(P)}$ indicated by P, the distribution over $v_{i(P)}$ conditioned on those realizations is still uniform. Of course, knowledge of v_i for $i \neq i(P)$ is useless for getting reward. Hence $\mathcal{A}$ can try at most $\beta k \log_2(t)/s = o(\beta k)$ maps in an attempt to find $M_{i(P)}$. Note that if we have a randomized algorithm which given a random P drawn from $\{0, 1, 2, \ldots, p-1\}^{n \times n}$ always outputs a set S of integers of size α such that $\mathbb{P}[i(P) \in S] \geq q$, then we can use it to construct a randomized algorithm that, given P, outputs an integer x such that $\mathbb{P}[i(P) = x] \geq q/\alpha$, simply by running the first algorithm and then selecting a random element of S. If $\mathcal{A}$ does not find $M_{i(P)}$, the distribution on the treasure's location is uniform given its knowledge. Hence it's budget of βk treasure locations can only earn it expected reward at most $\beta k/t$. Armed with these observations and Theorem 1.9 of Feige and Lund (1997) and our complexity theoretic assumptions, we infer $\mathbb{E}[R] \leq o(\beta k) \cdot 2^{-\ell}(1 + 1/n^\eta) + \beta k/t + 2^{-n^2}$. Since $s = n^3$ and $m = t = n^{1/\gamma}$ and $\gamma = \Theta(1)$ and $\eta = 1$ and $\ell = \log_2 m$ and $k = n^3 + s + 1 = 2n^3 + 1$, we have

$$\mathbb{E}[R] \leq \frac{\beta k}{t}(1 + o(1)) = 2\beta n^{3-1/\gamma}(1 + o(1)).$$

Next note that $|E| = t + ms + n^3 = n^{3+1/\gamma}(1 + o(1))$. Straightforward algebra shows that in order to ensure $\mathbb{E}[R] = o(\beta/|E|^{1-\varepsilon})$, it suffices to choose $\gamma \leq \varepsilon/6$. Thus, under our complexity theoretic assumptions, any polynomial time randomized algorithm $\mathcal{A}$ with budget βk achieves at most $o(\beta/|E|^{1-\varepsilon})$ of the value obtained by the optimal policy with budget k, so the approximation ratio is $\omega(|E|^{1-\varepsilon}/\beta)$. ■

A.7 A Symbol Table

E, $e \in E$	Ground set of items, and an individual item.
O, $o \in O$	States an item may be in, or outcomes of selecting an item, and an individual state/outcome.
Φ	A realization, i.e., a function from items to states.
Ψ	A partial realization, typically encoding the current set of observations; each $\Psi \subset E \times O$ is partial mapping from items to states.
$\sim$	The consistency relation: $\Phi \sim \Psi$ means $\Psi(e) = \Phi(e)$ for all $e \in \mathrm{dom}(\Psi)$.
$\mathbb{P}$	The probability measure on realizations.
π	A policy, which maps partial realizations to items.
$E(\pi, \Phi)$	The items selected by π when run under realization Φ.
$\Delta(e \mid \Psi)$	The conditional expected marginal benefit of e conditioned on Ψ: $\Delta(e \mid \Psi) := \mathbb{E}_\Phi[f(\mathrm{dom}(\Psi) \cup \{e\}, \Phi) - f(\mathrm{dom}(\Psi), \Phi) \mid \Phi \sim \Psi]$.
$\Delta(\pi \mid \Psi)$	The conditional expected marginal benefit of policy π conditioned on Ψ: $\Delta(\pi \mid \Psi) := \mathbb{E}_\Phi[f(\mathrm{dom}(\Psi) \cup E(\pi, \Phi), \Phi) - f(\mathrm{dom}(\Psi), \Phi) \mid \Phi \sim \Psi]$.
$\Psi[e/o]$	Shorthand for $\Psi \cup \{(e, o)\}$.
k	Budget on the cost of selected item sets.
$\pi_{[k]}$	A truncated policy. See Definition 4 on page 9 (unit costs) and Definition 26 on page 35.
$\pi_{[\leftarrow k]}$	A strictly truncated policy. See Definition 24 on page 35.
$\pi_{[k \rightarrow]}$	A laxly truncated policy. See Definition 25 on page 35.
$\pi @ \pi'$	Policies π and π' concatenated together. See Definition 28 on page 35.
f	An objective function, of type $f : 2^E \times O^E \to \mathbb{R}_{\geq 0}$ unless stated otherwise.
f_{avg}	Average benefit: $f_{\text{avg}}(\pi) := \mathbb{E}_\Phi[f(E(\pi, \Phi), \Phi)]$.
c	Item costs $c : E \to \mathbb{N}$. Extended to sets via $c(S) := \sum_{e \in S} c(e)$.
c_{avg}	Average cost of a policy: $c_{\text{avg}}(\pi) := \mathbb{E}_\Phi[c(E(\pi, \Phi))]$.
c_{wc}	Worst-case cost of a policy: $c_{\text{wc}}(\pi) := \max_\Phi c(E(\pi, \Phi))$.
c_Σ	Min-sum cost of a policy: $c_\Sigma(\pi) := \sum_{t=0}^{\infty} \left(\mathbb{E}_\Phi[f(E, \Phi)] - f_{\text{avg}}(\pi_{[\leftarrow t]})\right)$.
$c(\pi \mid \Psi)$	Conditional average policy cost: $c(\pi \mid \Psi) := \mathbb{E}_\Phi[c(E(\pi, \Phi)) \mid \Phi \sim \Psi]$.
α	Approximation factor for greedy optimization in an α-approximate greedy policy.
Q	Benefit quota. Often $Q = \mathbb{E}_\Phi[f(E, \Phi)]$.
η	The coverage gap: $\eta = \max\{\eta' : f(S, \Phi) > Q - \eta'$ implies $f(S, \Phi) \geq Q$ for all S and $\Phi\}$.
$\mathbf{1}_P$	The indicator for proposition P, which equals one if P is true and zero if P is false.

Table 2: Important symbols and notations used in this article